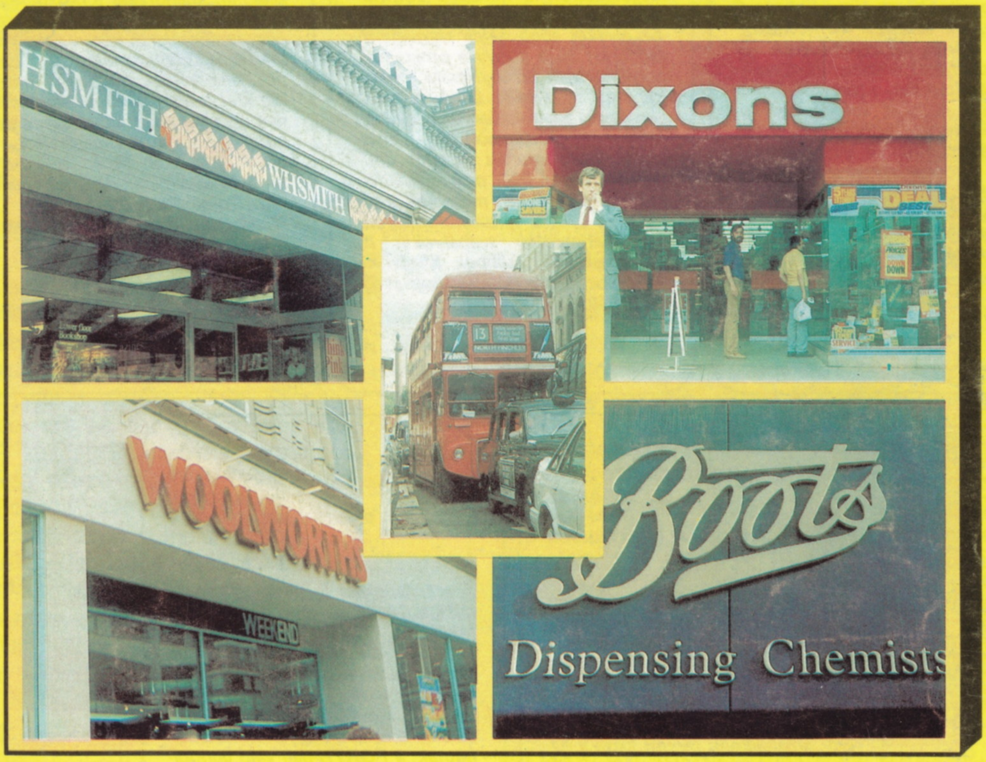
SPECTRUM

VILÁG 7. rész



a MŰSZAKI KÖNYVKIADÓ

könyvajánlata

DR. HARMATHA ANDRÁS:
HŐTECHNIKAI SZÁMÍTÁSOK SZEMÉLYI SZÁMÍTÓGÉPPEL

A könyv a legkülönbözőbb hőtechnikai számításokat tartalmazza, mindegyikhez közvetlenül működő programok listáját mellékelve. A programok az igen népszerű Spectrumra és a ZX 81-re készültek. A könyv használata rendkívül meggyorsítja az általában hosszadalmas számításokat, a tervezői, üzembentartói munkát sokkal hatékonyabbá teszi, továbbá elősegíti a számítástechnikai és energiagazdálkodási kormányprogramok megvalósítását.
A könyvhöz Spectrumra, és ZX 81-re írt programokat tartalmazó kazetták is készülnek.

240 old., fűzve 78,– Ft

SZLÁVI PÉTER–ZSAKÓ LÁSZLÓ:
MÓDSZERES PROGRAMOZÁS

A könyv, miután konkrét példák alapján megfogalmazza a programkészítéssel kapcsolatos problémákat, közérthető stílusban szól azok megoldásáról: a korszerű strukturált programozási módszerekről. Az elveket több példa szemlélteti. Módszert ad a könyv a programok helyességének bizonyítására, a hatékonyság növelésére is. A szerzők, elveiknek megfelelően egy elvont, magyar alapszavakból álló magasszintű programozási nyelvet definiálnak, ebben írják programpéldáikat, de függelékben megadják a különböző BASIC változatokra vonatkozó átkódolási szabályokat is. A függelékben, többek között, a ZX Spectrumra is találhatók programlisták.

116 old., fűzve 50,– Ft

PATAKI ATTILA–TALLÉR FERENC:
FŰTÉSI RENDSZEREK SZÁMÍTÁSA SZEMÉLYI SZÁMÍTÓGÉPPEL

A könyv segítségével elvégezhető a családi házak, társasházak fűtési rendszerének számítógépes méretezése, ill. ellenőrzése a hőszükséglet meghatározásától egészen a rendszer vezérléséig. A könyv ezen kívül ipari jellegű felhasználóknak is hasznos ismereteket nyújt (pl. kazánház-kialakítás). A könyv használata nem igényel számítástechnikai vagy hőtechnikai szakképzettséget, a programok több gépen való futtatásra is alkalmasak (ZX 81, Spectrum, Commodore 64).
Mindhárom gépre programkazetták is kaphatók (340,– Ft).

278 old., fűzve 158,– Ft

KEPES JÁNOS:
MIKROSZÁMÍTÓGÉPES GRAFIKA GRAFIKAI ALGORITMUSOK

A mikroszámítógépek látványos színei és nagy felbontóképessége szinte egy új műfajt teremtett: a számítógépes grafikát. A megjelenítésnek eddig sosem látott lehetőségei nyíltak meg, hiszen jól meghatározott utasítások hatásairól van szó, amelyeknek semmi köze a hagyományos eszközökhöz. A mű két oldalról közelíti meg a kérdést: egyrészt a matematikai képleteknek, programutasításoknak grafikai megjelenítését mutatja be, másrészt grafikai elemek matematikai, ill. számítástechnikai megoldását írja le. A könyv példái a ZX Spectrum BASIC nyelvén vannak megfogalmazva. Programkazetta is kapható hozzá! (kb. 300,– Ft.)

132 old., fűzve 48,– Ft

SERES–FENYŐ–GYALOGH:
A FORTH PROGRAMOZÁSI NYELV

A számítógépes nyelvek széles skáláján a BASIC térhódítása után leginkább a FORTH elterjedése várható. A nyelv jellegzetessége az igen kis tárigény és a rendszerprogramokban való jó használhatóság. Ez a kis kézikönyv, amelyet a kezdők és a haladók egyaránt használhatnak, a FORTH-ban való gondolkodásmódot ismerteti meg, tárgyalja a kompilerek jellegzetességeit. A függelék többek között tartalmazza a Spectrumra készült FORTH fordítók szintetikai leírását is.

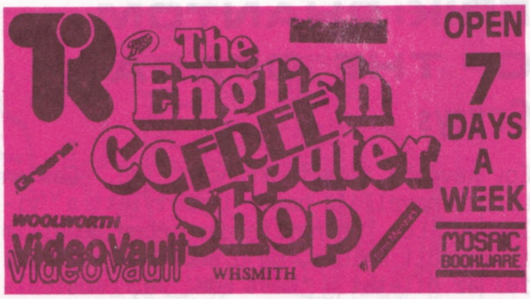
291 old., fűzve 84,– Ft

A könyvek megrendelhetők:

MŰSZAKI KÖNYVKIADÓ Kandó Kálmán Könyvesboltja
Budapest, V. Bajcsy-Zs. út 20. –1051

1. CSILLAPODÓ LONDONI LÁZ

Amikor egy Spectrumos először jár Londonban, a repülőtérre megérkezve repesve várja, hogy a hatalmas reklámtranszparensek között megpillantsa a Sinclair nevet, vagy valamely perifériát gyártó cég emblémáját. Sajnos csalódnia kell, ugyanis a töménytelen élelmiszer-, dohány-, italáru mellett sportkocsikínálat, HI-FI berendezések reklámai néznek farkasszemet a turistával. Elvétve fordul csak elő számítástechnika a falakon, de kivétel nélkül IBM PC és vele kompatibilis klónok gyártói kínálják portékáikat.
A repülőtér óriási csarnokát elhagyva többféleképpen is megközelíthetjük a Brit főváros centrumát. A tehetősebbek úri módra beülhetnek az évtizedek óta hagyományos AUSTIN taxi-ba, akik ráérnek, felülhetnek a városba tartó piros emeletes buszra, akik pedig gyorsan és viszonylag olcsón szeretnének utazni, igénybe vehetik a híres londoni metrót, melyet az 'őslakosok' egyszerűen csak TUBE-nak (cső) neveznek.
A városba érve újabb csalódás éri a lelkes Spectrumost. Nem érdemes úgy elindulni a belváros utcáin, hogy Sinclair számítógépekkel és tartozékaikkal foglalkozó üzletet keresünk, de még úgy sem, hogy kizárólag mikro-gépek software-eit elárusító boltba szeretnénk eljutni, mert előbb utóbb azon vesszük észre magunkat, hogy a Big-Ben toronyórája már az éjféli dallamokat játssza, ennek pedig egyenes következménye, hogy már csak néhány - kimondottan illatszereket és élelmiszert árusító - bolt tart nyitva, no meg a mulatók, éttermek is, de itt valószínűleg nem fog a pincér BATMAN-t kínálni nekünk féláron.
A lehetőségeinket teljesen más szemszögből kell felmérnünk. Londonban ill. az egész szigetországban óriási bolthálózatok és sok kis kereskedő együttes erővel küzd a fennmaradásért. Minden kereskedő jól meg'font'olja, mit tesz ki a kirakatba. Minden bolthálózatnak megvan a saját profilja. Az egyik legnagyobb monopólium a 'HARRODS', de ők sem részesítik előnyben a Sinclair gépek és tartozékaik árusítását. Több nagyobb bolthálózat (pl. DIXONS, W.H.SMITH) azt árulja, amire éppen igény van, pl. jelenpillanatban náluk már 16/48K gép nem is kapható, csak a 128K különböző módozatait (+, +2, +3) árulják. A perifériaellátottság sem töké-



letes, a vásárlóerő igen rövid idő alatt nagy mennyiséget szívott fel, és a kereslet egyszerűen visszaesett. Ebből adódóan MULTIFACE-re, vagy OPUS-ra elég nehezen bukkanhatunk. A speciális perifériákat inkább kisebb műszaki kereskedésekben érdemes keresni (pl. K.K.STATIONERS). Ezekben általában 'hegyekben' állnak a kicsomagolatlan dobozok, csak a vásárló szerencséjén múlik, hogy éppen talált olyat, ami érdekli (pl. 3 csatornás hanggenerátort).
Ugyanez a 'pangás' érezhető a software forgalmazással kapcsolatban is. Software is csak a kijelölt bolthálózatokban kapható (pl. WOOLWORTH, LASKYS, BOOTS, RUMBELOWS, W.H.SMITH, COMET, stb.). Nem kell arra számítani, hogy ezekben a boltokban minden programot megvehetünk, amit szeretnénk, pl. a WOOLWORTH hálózat boltjaiban a hanglemezek, audio- és video-kazetták ill. CD lemezek mellett találjuk meg a polcokon a software-eket, de itt csak az olcsóbb (1-2 Font értékű) MASTERTRONIC, CODE MASTERS stb. programok kaphatók. A W.H.SMITH-nél már jobb a választék, de drágább, és felhasználói software itt sem kapható. Több napi csatangolást követően mindenki belátja, hogy az angolok főképpen megrendelésre vannak berendezkedve, nem is csoda, hogy software házak tömkelege kínálja portékáit a folyóiratok hasábjain, hiszen érdeke, hogy csak annyit sokszorosítson, amennyire igény mutatkozik. Mindettől függetlenül, ha összehasonlítjuk a jelenlegi helyzetképet az öt évvel ezelőttivel, megállapíthatjuk, hogy csillapodott a londoni Spectrum láz, az akkori fiatalok többsége ma már AMSTRAD, vagy ATARI gépre pártolt át, igaz újabbak vették át a stafétabotot, mind a fejlesztők, mind a felhasználók körében.
Mi azért reméljük, hogy a hazautazó Spectrumos nem fog rossz szájízzel távozni a szigetországból, még akkor sem, ha megmaradt 2 Fontján a repülőtéren már csak csokoládét vehet, pedig a FEUD-ot 1,99-ért árulták a PICCADILLY mellett.

2 2. JÁTÉK, POKE, TÉRKÉP

ERIK: PHANTOM OF THE OPERA

Crysys

Egy kevésbé ismert francia író, Gaston Leroux novellája hamar ismert lett a világban. Lon Chaney rendezésében némafilm is készült belőle, majd Webber használta fel az ötletet a 'West End' c. musicalben. Most a Crysys piacra dobta a történet számítógépes változatát. Erik egykor csinos fiú volt, de egy szerencsétlen tűz következtében borzasztóan csúnya lett, és együgyű. Bosszúvágyat érzett a szépséggel megáldott emberek irányában, így esett meg, hogy a párizsi operaház ismert szólistáját, Krisztinát elrabolta, és elrejtette az operaház alatt húzódó alagútrendszerben. Raoul a tenorista Krisztina keresésére indul, de a labirintus bonyolult, és tele van veszélyekkel. Izgalmas történet, jó grafika.

DEVIANTS

Players

Az akciójátékok kedvelőinek ajánljuk ezt a 'V' stílusú játékot. A 'DEVIANTS' hívei birtokukba vettek egy kulcsfontosságú űrállomást. A csillagközi rendőrség osztagai régóta azon fáradoztak, hogy 'kitakarítsák' az űrállomást, de nem sok sikerrel jártak, ezért elhatározták, hogy megsemmisítik azt. A bevetéshez szükséges eszközöket titokban becsempészték a bázisra, és főhősünknek kell a feladatot végrehajtania. Meg kell találni a beprogramozatlan nukleáris bombát, valamint annak mind a 6 db. detonáló rúdját. A detonáló rudakhoz darabonként 6 db. megfelelő kulcsot is össze kell gyűjtenünk, s természetesen néha 'megzavarnak' bennünket ténykedésünkben. Feszült izgalom, szép sprite-ok, a zene lehetne jobb is.

SIDEWALK

Infogrames

A francia illetőségű software ház sem tétlenkedett az utóbbi időben. A 'Sidewalk' (járdaséta) humoros történettel fűszerezett kalandjáték. Történetünk hőse pórul járt, ugyanis a motorját ellopták. Ez kínosan érintette őt, ugyanis szorítja az idő, mivel barátnőjével koncertre készültek az este, és még a jegyet sem volt ideje megvenni. Elindul tehát a városban, hogy megtalálja a motort, mert csak a segítségével tudja az esti programot megszervezni. A képernyő felső része mutatja a város utcáit, alatta üzenetmező, legalul pedig az un. ikon-mező található. Az utcán vannak jó és rossz emberek, ez utóbbiakkal esetleg verekednünk is kell. Energiaszintünket a jobb oldalon látható söröskrigli jelzi.

STAR PILOT

Firebird

Nem áll szándékunkban ócsárolni egyetlen cég játékát sem, főként akkor nem, ha tudjuk, hogy a Firebird és a Mastertronic havonta 4-5 játékkal is meglepi a piacot. Ettől függetlenül 1988. elején megjelenni egy olyan játékkal ami még 1984-ben is elég kemény kritikát kapott volna, elég nagy bátorság. A játék kezelése rendkívül egyszerű, és stílusára megannyi 'klónt' fedezhettünk fel az utóbbi időben. Egy folyamatosan előre haladó űrhajóval kell monoton lövöldöznünk előre, miközben balra-jobbra navigálással kerüljük ki a felismerhetetlen alakú korlátokat és a repkedő tárgyakat. Ha ezt a szintet teljesítjük, bejutunk az űrbázisra, melynek 'gyenge' pontjára bombát kell ejtenünk. Közepes grafika, hiányzó ötletek.



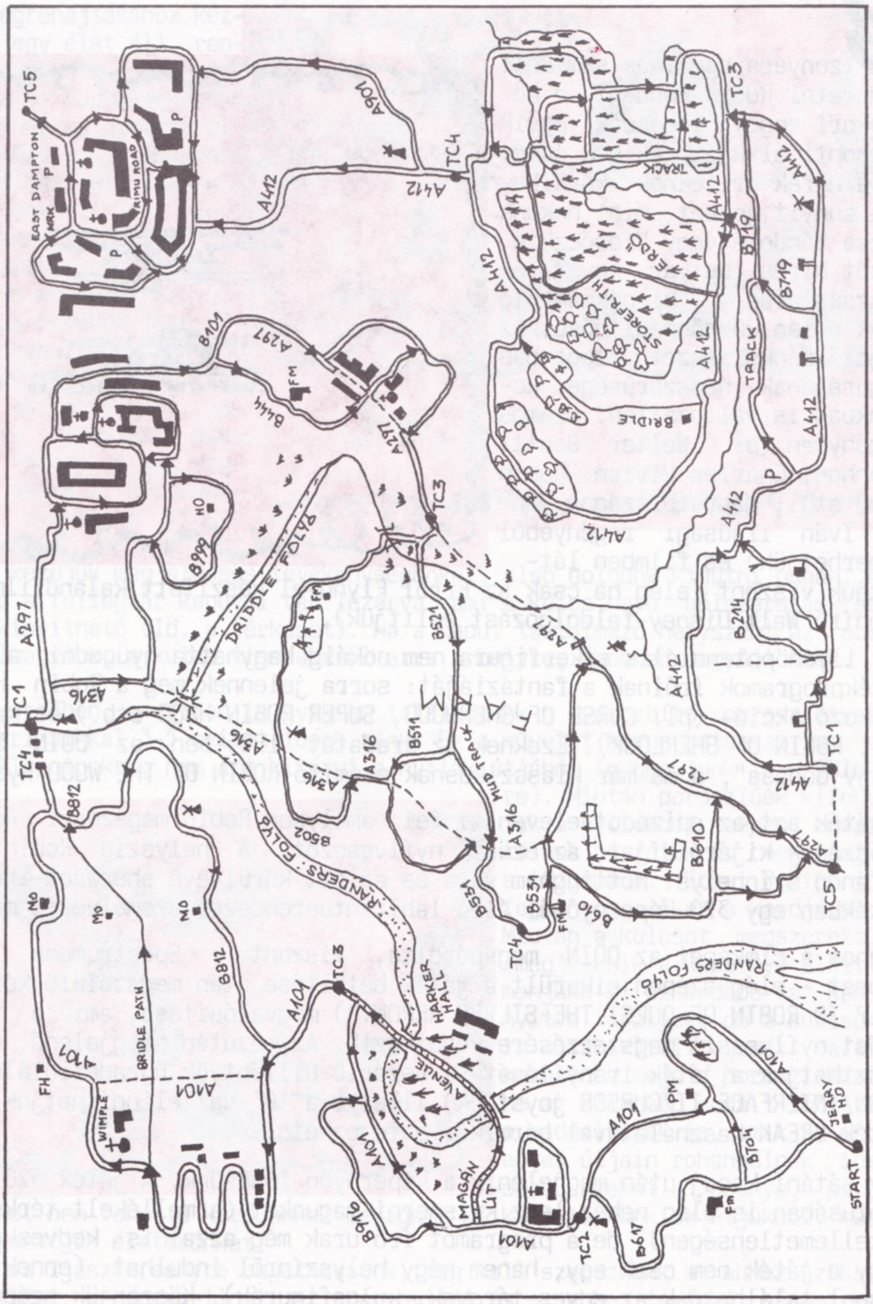
Az örökélet csak akkor hatásos, ha a program a BASIC-ből, és egy 48151 byte-os részből áll.
Töltsük be a LOADER-t, és állítsuk le a magnetofont. Ezután írjuk be a következő BASIC programot:

```
10 CLEAR 25999: FOR i=65007 TO 65070: READ a: POKE i,a: NEXT i:
   RANDOMIZE USR 65007
20 DATA 49,255,255,221,33,0,64,17,23,188,62,255,55,205,86,5,33,88,39,
   217,175,237,79,33,20,254,17,0,91,1,30,0,237,176,195,0,91,49,254,102,
   251,33,189,143,175,119,35,119,35,119,33,23,252,17,255,255,1,0,153,
   237,184,195,161,181
```

Futtassuk a programot (RUN+ENTER) és indítsuk el a magnót.

2. JÁTÉK, POKE, TÉRKÉP

RALLYE DRIVER térkép



Ranarama

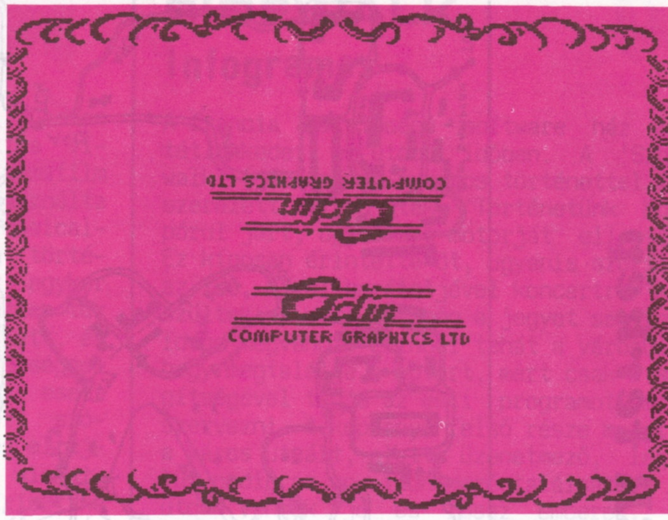
Az örökélet csak akkor működik ha a program a következő részekből áll: 170/35872/1399/1968

Töltsük be a LOADER-t, és RESET-eljük a gépet, majd írjuk be a következő BASIC programot:

```
10 CLEAR 24791: LOAD"" CODE: LOAD"" CODE: RANDOMIZE USR 24830:
   LOAD"" CODE 16464
20 POKE 57436,205: POKE 59836,205: POKE 57572,201: POKE 59821,0:
   POKE 57421,0: RANDOMIZE USR 24833
```

Futtassuk a programot (RUN+ENTER) és indítsuk el a magnót.

2.1 ROBIN OF THE WOOD



Bizonyára senkinek sem kell bemutatni Robin Hood-ot, a középkori angol legendák egyik központi alakját, akinek a neve a derék briteknek körülbelül annyit jelent, mint nekünk Rózsa Sándoré vagy Vidróczkié, a két híres betyár neve: az igazságtalan, kizsákmányoló urak ellen küzdő népi hősé. Az angol népköltészet legendás figurájának népszerűsége korunkban is változatlan: számos regényben (pl. Walter Scott: Ivanhoe; Charles Vivian: Robin Hood stb.; Magyarországon Mándy Iván ifjúsági regényéből ismerhetjük) és filmben láthatjuk viszont (elég ha csak az Errol Flynn-el készített kalandfilmet vagy a világhírű Walt Disney-feldolgozást említjük).

Egy ilyen potenciális sikerfigura nem sokáig hagyhatta nyugodni a számítógépes játékprogramok íróinak a fantáziáját: sorra jelennek meg a Robin Hood-dal foglalkozó akció- (pl. CURSE OF SHERWOOD, SUPER ROBIN HOOD stb.) és kalandprogramok (pl. ROBIN OF SHERLOCK). Ezeknek az áradatát 1985-ben az ODIN Software első "nagy dobása", a ma már klasszikusnak számító ROBIN OF THE WOOD nyitotta meg.

A játék azt az epizódot eleveníti fel, amelyben Robin megszerzi Anglia legjobb íjászának kijáró díjat, az ezüst nyílvesszőt. A helyszín Robin kalandjainak állandó színhelye: Nottingham vára és az azt körülvevő sherwood-i erdő, amit a játékban egy 320 képernyőből álló labirintusrendszer személyesít meg.

Sajnos a címképet az ODIN megspórolta, viszont - Spectrumunk lehetőségeihez képest - elég szépen sikerült a játék betöltése után megszólaló kérdés ("CAN YOU HELP TO ROBIN OF QUEST THE SILVER ARROW?") megvalósítása, ami a fent említett ezüst nyílvessző megszerzésére vonatkozik. Az ezután megjelenő menüben kiválaszthatjuk a játék irányítását (a vezérlő billentyűk újradefiniálása vagy KEMPSTON/INTERFACE II/CURSOR joystick) illetve a 'Ø'-val elindíthatjuk a játékot. A játék BREAK használatával bármikor befejezhető.

Egy sátáni kacaj után megjelenik a képernyőn főhősünk. A játék 320 szobás labirintusában is elég nehéz lesz kiismerni magunkat (a mellékelt térkép segít ezen a kellemetlenségen), de a programot író urak még azzal is kedveskedtek nekünk, hogy a játék nem csak egy, hanem négy helyszínről indulhat (ennek függvényében máshol találhatóak az egyes tárgyak, kulcsfigurák). Köszönjük szépen!

Batty

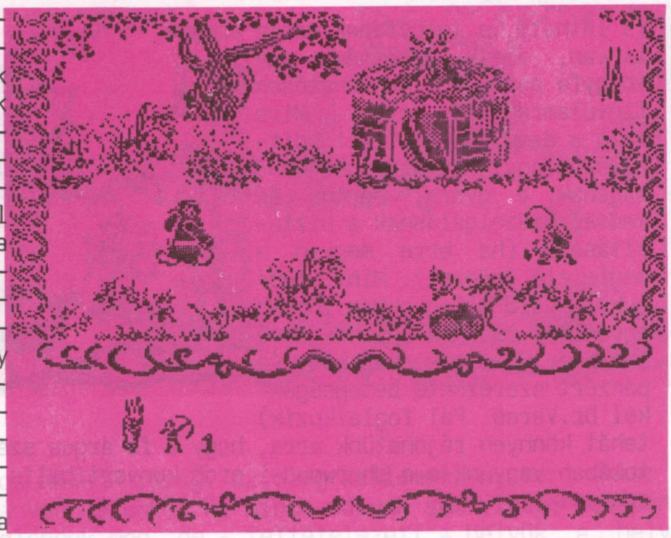
Az örökélet beviteléhez a következő file-térkép szükséges: fejléces kód, fejléces SCREEN, fejléces kód.
Töltsük be a LOADER-t és állítsuk meg a magnót, RESET-eljük a gépet majd írjuk be a következő betöltőt:

```
10 CLEAR VAL"23999":LOAD"" CODE:LOAD""SCREENØ
20 LOAD"" CODE VAL"24000"
```

Futtassuk a programot és indítsuk el a magnetofont. Ha betöltött minden részt, írjuk be a következő parancsokat:

```
POKE 47864,0 (ENTER)
RANDOMIZE USR 65335 (ENTER)
```

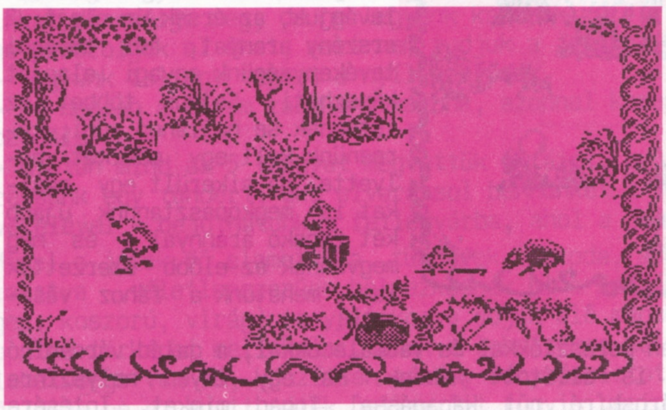
2.1 ROBIN OF THE WOOD



A feladat végrehajtásához kezdetben csak egy élet áll rendelkezésünkre, de a poroszlók nyilainak és a vadmalackáknak köszönhetően ez is gyorsan elfogy. Életerőnk a játékképernyőt alulról határoló ágas-bogas "izé"-n mérhető le: minél sötétebb, annál közelebb a vég. Jól tesszük, ha az erdőben található koszorúk begyűjtésével és az erdőben lakó bácsikánál összeszedünk néhány életet. A bácsika az első találkozásnál ad egy életet, ezután már csak elveszett életünket adja vissza. A még meglevő életek száma a játékképernyő alatt, a nyilazó figura mellett látható.

Ha a várban kezdjük meg áldásos tevékenységünket, első dolgunk kijutni onnan. Az alsóváros és a fellegvár kapuval van lezárva, ami a kulcs nevű célszerszám segítségével kinyitható (ld. a térképet). Ha a kaput tartalmazó helyszínt az erdő irányában elhagytuk, a kulcs visszakerül arra a helyre, ahonnan begyűjtöttük.

Akárcsak az erdőben, a várban is gyakran találkozunk a városbíró poroszlóival, akik számszeríjaikkal 3-5 nyilat lőnek ránk. Ez a művelet káros hatással van az egészségi állapotunkra, nem árt kitérni a nyilak útjából (guggoljunk le a földre). Miután poroszlóék kifogytak a nyilakból szépen odasétálunk hozzájuk és megnyomjuk a tűzgombot. Erre Robin jól fejbenyomja őket a kardjával. Miután a kulcsot megszereztük (megjelenik a képernyő alatt), menjünk ki az erdőbe (a kaput nyitva találjuk, de a kulcsunk is eltűnik). Itt ismét találkozhatunk majd a poroszlókkal, de az ellenség sorait bővítik a vadmalackák is, amik az erdő egyes útjain rohangálnak ideoda. Ezeket a csúnya, piros teremtményeket nem lehet a kardunkkal levágni, ki kell térnünk az útjukból, mert gyorsan fogyasztják életerőnket.

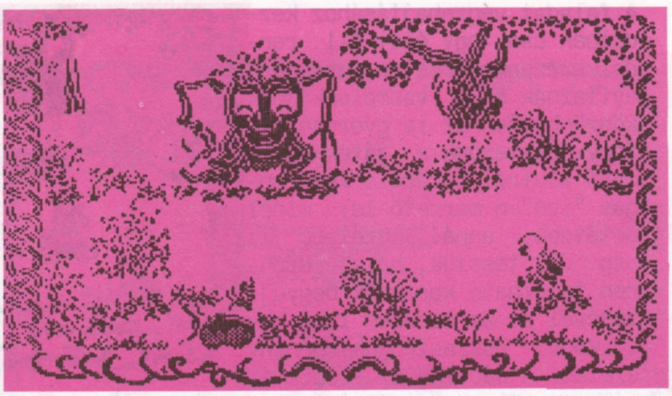


Ahhoz, hogy az íjászversenyre bejuthassunk, meg kell szereznünk az ehhez szükséges felszerelést: a csodapallost, az aranynyilat és az ehhez tartozó három varázs-nyílvesszőt. Ezek megszerzése a Vas- és Edénybolt helyi képviseleténél történhet, amelyet különös módon egy öreg tölgyfában helyeztek el. Mint az a képen

C.O.R.E.

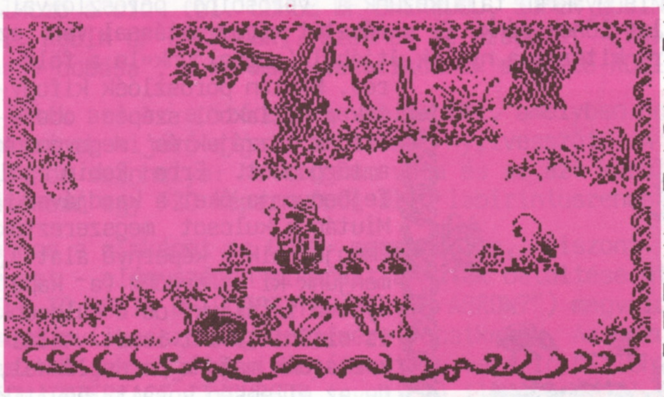
```
File-térképe a következő: 164/6912/20000/20536/6916/1704.
A végtelen élethez töltsük be a LOADER-t, BREAK, majd írjuk be:
INK 7 (ENTER) 20 REM (ENTER) POKE 23757,96 (ENTER)
Írjuk be a következő BASIC programot:
30 FOR i=23841 TO 23853:READ a:POKE i,a:NEXT i
40 DATA 62,0,50,216,157,62,255,55,229,195,86,5,0
és futtassuk le (RUN+ENTER), majd indítsuk el a gépi kódú betöltőt a
RANDOMIZE USR 23760 (ENTER) paranccsal.
```

2.1 ROBIN OF THE WOOD



is látható, a tölgyfának szeme is van, amely alkatrész egy tölgyfa esetében meglehetősen figyelemreméltó dolog. Mire való a szem? Az említett bolthálózat egyik kiváló tulajdonságának, a gyors, pontos és kedves kiszolgálásnak a biztosítására (ha erre megvan a megfelelő fedezet). Mint tudjuk, kiváló világunkban ingyen egyenlőre csak a nátha nevű közismert betegség kapható (a pénzért szerezhető betegségekkel Dr.Veres Pál foglalkozik) tehát könnyen rájöhetünk arra, hogy a fa árgus szemei azt vizsgálják, hogy birtokában vagyunk-e a sherwood-i erdő konvertibilis fizetőeszközének, a 3 zacskó aranynak.

Robin-nak sajnos nem volt meg az a lehetősége, hogy maszek zöldségkereskedést nyisson (tőmondatban: nincs pénze), ezért pénzt csak az útonállás (majd úton való leütés) néven közismert módszerrel tud csak szerezni. Erre van rendszeresítve egy-egy érsek bácsi, akik - meglehetősen könnyelmű módon - némi egyházi adománnyal a sherwoodi erdőben mászkálnak le-fel. Egy-egy érsek kíséretébe tartozik két poroszló is, akik buzgón lövöldöznek számszeríjaikkal mindenre, ami csak körülöttük látható. Ha a lövöldöző urakat levágjuk, az érsek úr elejt 2 erszény aranyat, ami további tevékenységünk anyagi alapját képezheti. Sajnos többet az istennek se hajlandó adni, keresnünk kell egy másikat helyette. Ha sikerült egy másikat is megkopasztanunk újabb két zacskó arannyal - és még megvannak az előbb szerzettek is -, mehetünk a fához vásárolni.



Nem elég, hogy az erdő tele van poroszlókkal és vadmalacokkal, a galád városbíró az erdő egyes részein csapdát is állított nekünk. Ha egy ilyen helyszínre besétálunk, a középen álló városbíró rút kacagással fogad minket. Intésére bemasírozik két poroszló és belénkereszt 8-10 nyilat. Mozogni ilyenkor már nem tudunk, egy életünk biztosan odavész. Ha ez az utolsó volt, akkor a GAME OVER nevű jól ismert feliratot láthatjuk, ha még volt életünk, akkor a várban lévő startpozícióba kerülünk.

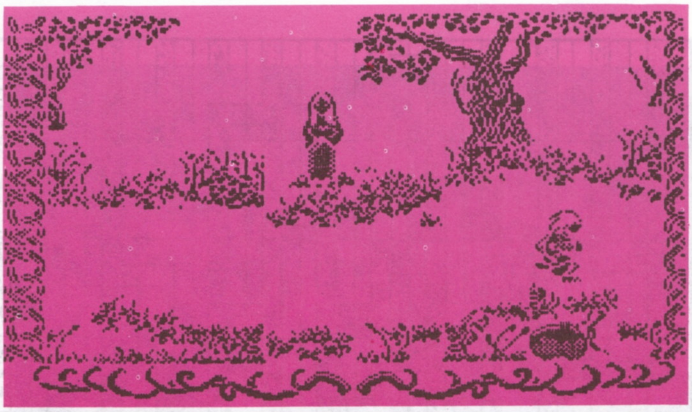
Paperboy

A sérthetetlenséghez feladatunk: a 48023-as címen, 255-öt elhelyezni.
Csak akkor működik, ha a file-térkép a következőképpen néz ki: 169/6912/20000/20536/6916/1704
Töltsük be a LOADER-t: MERGE"", majd írjuk be a következő BASIC programot:

```
30 FOR i=23862 TO 23869: READ a: POKE i,a: NEXT i
40 DATA 50,151,187,55,229,195,86,5
```

Futtassuk a programot (RUN+ENTER) és indítsuk a magnót.

2.1 ROBIN OF THE WOOD



Erdőbeli kalandozásainkat néhány tündér is szemmel követi, néha sikerül beléjük botlanunk. A tündérek különös érdeklődést tanúsítanak a botanika iránt, mert a velük való találkozás a főhősünknél lévő virágmennyiség szerint lesz jó illetve rossz kimenetelű: ha egy virág van nálunk, akkor csak szuvenírként elveszik; ha két virággal rendelkezünk, maximumra növelik életerőnket; ha pedig három virág van nálunk, elvarázsolnak bennünket a - térképen szaggatott vonallal jelölt - várkertbe. Ha nincs nálunk virág, az elég kellemetlen, mert akkor a pénzeszacskóinkat veszik el (ha van nálunk). Ha nincs nálunk sem virág, sem pénzeszacskó, a tündérek nem is jelennek meg, de - feltételezett - helyük a helyszínen lévő sövénysoron látható jellegzetes bevágásról felismerhető.

Ne szaporítsuk tovább a szót, tekintsük át inkább a játék folyamán követendő stratégiát:

- mindenekelőtt gyűjtsünk minél több életet, abból baj nem lehet (ha elvesztettünk egyet, mindig térjünk vissza az apókához, hogy visszakapjuk)
- keresgéljünk az erdőben érsekeket és raboljuk ki őket (némi időbe telik, talán pár óra kérdése az egész)
- mindig legyen nálunk legalább egy virág, hogy a tündér nehogy a pénzeszacskóinkat dézsmálja meg, de ne legyünk kapzsik: a virágok közül csak néhány nő újra, ne pazaroljuk el őket, mert nem tudjuk a játékot befejezni
- vegyünk fel egy tegezt a nyílvesszőnek
- ha megvan a három zacskó aranyunk, azonnal keressük meg a fát és vásároljunk be nála
- miután mind az 5 tárgyat megvettük a fától, gyűjtsünk össze 3 virágot és keressünk egy tündért, aki elvarázsol bennünket a várkertbe. Innen - a kulcs felvételével - bejuthatunk a fellegvárba, ahol a jelzett ajtón kell bemennünk...

Mivel a játék fő elemei (fa, tündérek, csapda, apó) - néhány pozíció között választva - véletlenszerűen helyezkednek el a pályán és a felveendő tárgyak (tegez, koszorú, virág) is váltakozva adják elő magukat, sajnos nem tudunk hajszálpontos térképet mellékelni, csak a lehetséges helyeket tudtuk bejelölni. Akinek lehetősége van rá, készítsen néhány fénymásolatot a térképről és játék közben jelölje be az adott játék jellemzőit (úgyis végigjárja néhányszor az összes szobát). Ha a játék véget ér, a skiccét el is dobhatja, ugyanilyen elrendezést úgysem kap még egyszer (megjegyeznénk, a feltételes jelölések sem az összes lehetséges pozíciót jelzik).

A térképen ponttal jelöltük a felveendő tárgyak (nyíl, koszorú, tegez) pozícióját; "A"-val az apó lehetséges helyeit (minden játékban csak egy van); "F"-fel a fa lehetséges helyeit (minden játékban csak egy van); "CS"-vel a csapdát (minden játékban csak egy van); "T"-vel a tündér lehetséges helyeit; "S"-sel a startpozíciókat. Az érsekeket nem tudjuk jelölni, ezek véletlenszerűen jelennek meg.

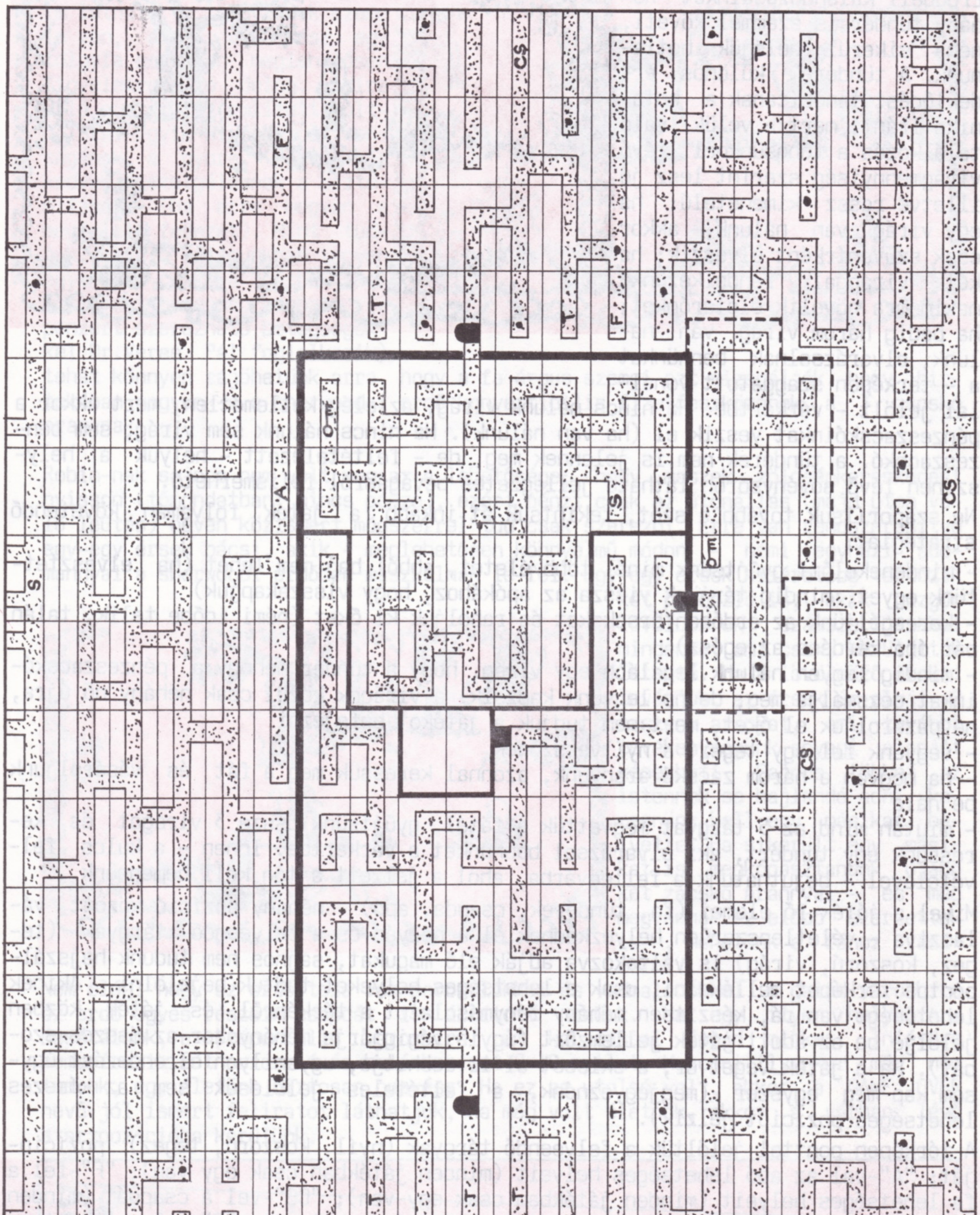
Loco

A program felépítése a következő: BASIC/fejléces CODE. A végtelen élethez töltsük be a LOADER-t, majd RESET-eljük a gépet. Írjuk be a következő programot:

```
10 CLEAR 24999: LOAD"" CODE: POKE 37094,0: RANDOMIZE USR 33038
```

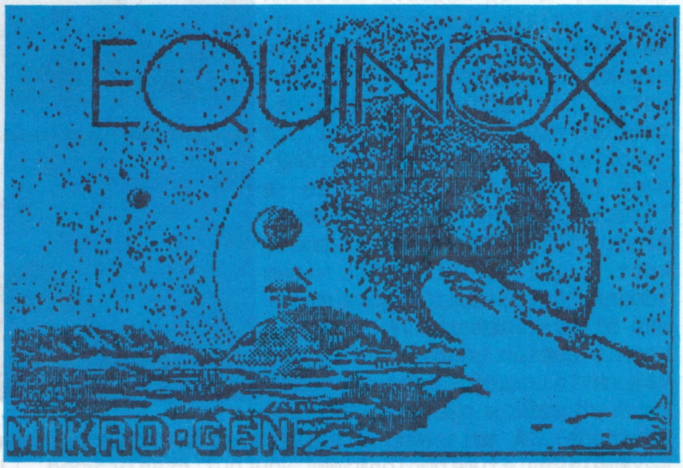
Futtassuk a programot (RUN+ENTER) és indítsuk el a magnót.

2.1 ROBIN OF THE WOOD



2.2 EQUINOX

A nagysikerű Wally-sorozat tagjaival szemben méltatlanul mellőzött játék a MIKRO-GEN 1986 elején megjelent EQUINOX című programja. A mellőzöttség érthetetlen, hiszen az EQUINOX is magán viseli a Mikro-Gen jellegzetességeket: egy labirintus rendszerben szétszórt tárgyakat össze kell gyűjteni és bizonyos logika alapján felhasználni. Talán csak az szólhat a játék ellen, hogy sokkal nehezebb, mint a Wallysak.



Az EQUINOX a végtelen űr egyik távoli bolygóján játszódik. A SURY-ANI naprendszer 7. bolygóján már felfedezése óta bányatelep üzemel, folyamatosan folyik a radioaktív anyagokat tartalmazó kőzet kitermelése. A kőzetből nyert sugárzó anyagot a későbbi energiahordozóként történő felhasználás céljából speciális tartályokban helyezik el. Egy napon hatalmas meteorit csapódik be a bolygóba és megrongálja a felszín alatt húzódó kiterjedt bányarendszert. A detonáció következtében szétszóródnak a tartályok. A lakosság - a tartályokból felszabaduló energiától való félelmében - elmenekül a bolygóról és csak egy szuper-robotot (a játékost) hagynak itt, hogy a bányászok által itthagyott szerszámok és tárgyak felhasználásával hordja össze a szétszóródott tartályokat az eredeti helyükre, mielőtt azok még csinos gombafelhővé nem változtatják a bolygót.

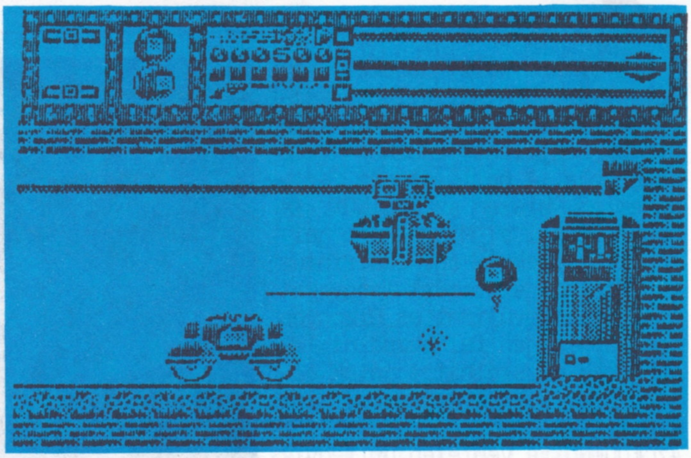
A bányakomplexum nyolc, egymástól jól elkülöníthető egységből (szintből) áll. Minden szinten 16 szobát és szintenként egy-egy sugárzó anyagot tartalmazó tartályt találhatunk, amelyet el kell helyeznünk a vákuumcsőbe, ami továbbítja azt rendeltetési helyére, a sugárvédett, ólomfalakkal lezárt szoba polcaira. A szintek közötti átjárást teleportáló kártyák biztosítják.

A meteorbecsapódásból kifolyólag elnéptelenedett bolygót csak ronda mutáns élőlények lakják, amelyek előszeretettel tartózkodnak a bányában. Ezek állandóan támadnak bennünket, ha túl sokszor vagy túl sokáig érintkezünk velük, egy idő után elvész egy életünk. Robotunk a mutánsok ellen egy lézerágyúval van felszerelve, amivel megsemmisíthetjük a túl közel merészkedő urakat. Más lehetőség is kínálkozik az élővilág eme kevésbé jeles képviselőinek likvidálására: a szintek egyes szobáiban bombákat találhatunk, amelyeket felvéve majd aktivizálva elhalálozásra kényszerítjük az aktuális szoba élőlényeit. A mutánsoktól mentes közlekedést a - térképen az adott szobák között nyíllal jelölt irányban - egy speciális mágneses lift biztosítja, de sajnos ez csak kevés helyen található.

A játék bejelentkezés után zenélget nekünk egy keveset, majd a főmenü opcióihoz tartozó számbillentyűk segítségével beállíthatjuk az irányító billentyűket ('1'), rövid - egyben angol nyelvű - használati utasítást kérhetünk ('2'), joystick-ot választhatunk ('4') vagy elindíthatjuk a játékot ('3'). Ha nem az említett billentyűk valamelyikét nyomjuk meg, rövid időn belül elindul a demo, amely bemutatja a játék fontos tárgyait (vákuumcső, teleportáló terminál stb.) és megvizsgálhatunk néhány szobát is.

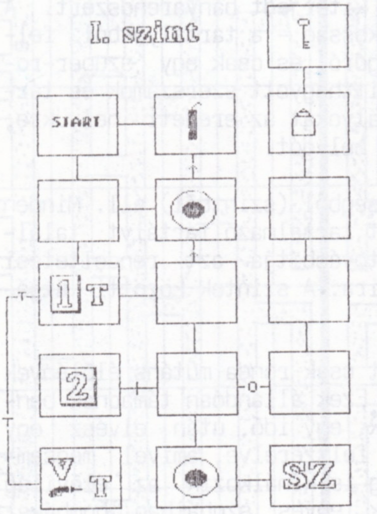
A játék képernyőjének bal felső sarkában látható azon tárgyak képe, ami nálunk van. Tárgyakat azzal a billentyűvel tudunk felvenni illetve használni, amelyet a billentyűzet definiálásánál a 'USE' kérdésre választottunk. A felvett tárgyakat nem tudjuk letenni, csak felhasználáskor, vagy akkor tűnnek el, ha másik tárgyat veszünk fel helyette (csere).

2.2 EQUINOX



A tárgy képe mellett látható robotok a még meglévő életeink számát jelzik (kezdetben 2 db. és amivel pillanatnyilag játszunk). A robotok mellett láthatjuk az eddigi tevékenységünkkel elért pontszámunkat. A jobb felső sarokban két mérőműszer található: a tolóerő és a lézertöltet kijelzői. Ezeket mindig kísérjük figyelemmel, mert a töltet kifogyása (nem tudunk lőni), de főleg a tolóerő hiánya (nem tudunk felemelkedni) esetén könnyű prédájává válunk a mutánsoknak (vagyis bekövetkezik a gyors halál nevű történés). A tolóerőt és a töltetet a szobákban szétszórt tartályok használatával maximálisra 'tuningolhatjuk'.

A következőkben ismertetjük az első három szint teljesítésének menetét. Az akció leírása a mellékelt térképeken alapul, célszerű az olvasással párhuzamosan a térképen is nyomon követni az események menetét.



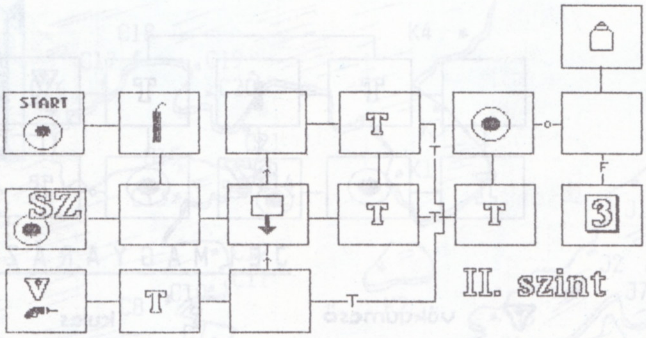
A játék elindítása után a startszobában vagyunk. Menjünk le az alsó soron lévő teleportáló korongért. Ezek a korongok a teleportáló terminálnoknál bekapcsolva engedélyezik a terminálpár kétszeri használatát (pl. oda-vissza utazás). Térjünk vissza tehát a koronggal a teleportáló helységbe. A teleportáló szerkezetbe beállva aktivizáljuk a korongot. Ha ezután hozzáérünk a masina tetejéről lelógó gömbhöz, - némi hangzavar után - egy másik teleportálóban találjuk magunkat. A vákuumcső másik oldalán egy fúrót találunk. Vegyük fel és teleportáljunk vissza. A fúróval menjünk fel a dinamittal jelzett szobába. A dinamit kultúráltan be van zárva a páncélszekrénybe, de a fúróval ezt kinyithatjuk és felpakolhatjuk a durrogó rudakat. Ezzel ballagjunk le a második sor középső szobájába, ahol a jobbra lévő kijáratot egy kőomlás zárja le. Aktivizáljuk a dinamitrudakat és máris idézhetjük az Equinox-rajongók legrégibb, írásban is fennmaradt mondását: "látjátok feleim szümtükhel, isa már csak pur és chomu van, omlás már csak vala..."

Miután bejutottunk, repkedjünk felfelé a tartályig, vegyük fel, majd az alatta lévő szinten indexeljünk be balra és cseréljük ki a másik teleportáló korongra. Ezzel menjünk el a teleportáló terminálhoz és aktivizáljuk. Még ne teleportáljunk, majd csak azután, hogy elhoztuk a tartályt. A teleportálás után aktivizáljuk a vákuumcsőben lévő nyílásban a tartályt. Egy szoba felvillanó képe jelzi, hogy AZ ELSŐ TARTÁLY A HELYÉRE KERÜLT!

Teleportáljunk vissza, majd menjünk fel a kulcsos szobába. A kulcs nyitja az egykori kőomlással szemben található ajtót, amely mögött felvehetjük a 2-es számú kártyát. Ennek birtokában tudunk csak átjutni a nagy teleporton keresztül a játék II. szintjére. A nagy teleport a jobb alsó szoba végében található, belépve a szint jelénél aktiválva a szinthez tartozó (és nálunk lévő) kártyát, utazhatunk a következő szintre. Menjünk át rajta keresztül a II. szintre.

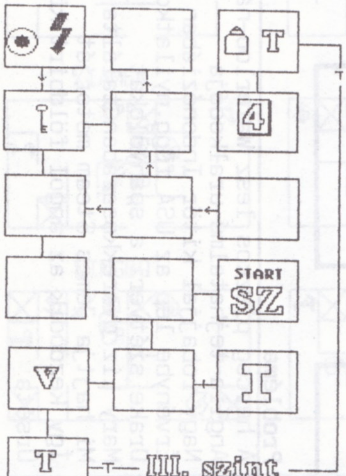
Az I. szintről a bal felső sarokban jelzett szobába érkezünk, ahol a 2-es kártyát egyből ki is cserélhetjük a teleportáló korongra. Ezzel menjünk el a T1 jelzésű teleportálóba és teleportáljunk egyet. Megérkezés után menjünk át a balra lévő szobába

a fúróért és vegyük fel, ezután menjünk át a teleport másik oldalán lévő szobába, szálljunk be a mágnesliftbe, utazzunk fel egy emeletet és balra kiszállva visszamegyünk a következő teleportáló korongért. Ez a trükk azért szükséges, mert így az előbb igénybevett terminálon még egy teleportálást hajthatunk végre anélkül, hogy újabb korongot kellene felhasználnunk.



A startszobában a fúróval kinyithatjuk azt az ajtót, ami elválaszt minket a teleportáló korongtól és felvehetjük. Vigyük el ezt az előbb igénybevett terminál alatt lévő szobába, ahol egy másik terminált találunk. Helyezzük bele a korongot, de ne teleportáljunk, hanem előbb hozzuk el a dinamitot. A robbantócuccal felszerelve már elvégezhetjük a teleportálást és az érkezési terminál feletti szobában robbantsuk ki a kőomlást. Még ne menjünk át, hanem a teleportáló korongot vigyük be az érkezési terminálba (de ne teleportáljunk!), hogy vissza tudjunk menni. Ezután elmehetünk a tartályért, majd teleportáljunk vissza. A tartállyal felmehetünk a felettünk lévő szobába, ahol a terminál még egy utazás erejéig aktív, tehát még teleportálhatunk vele egyet. A teleportálás után a tartályt vigyük át a bal oldali szobában lévő vákuumcsőhöz és tegyük bele a tartályt. A védőszoba felvillanó képe jelzi, hogy MÁR KÉT TARTÁLYT TETTÜNK VISSZA A HELYÉRE!

Szálljunk be a mágnesliftbe és egy emeletet utazva vegyük fel az ásót, amivel a jobb oldal felé lévő teleportáló terminálon keresztül utazunk egyet. Megérkezés után meg kell szereznünk a 3-as számú kártyát, ami egy földrakással elzárt átjáró mögött található. A mikroásó aktivizálásával az akadályt megszüntetve bejuthatunk és felvehetjük a 3-as kártyát. A teleportálón keresztül térjünk vissza a nagy teleporthoz és lépjünk tovább a III. szintre.

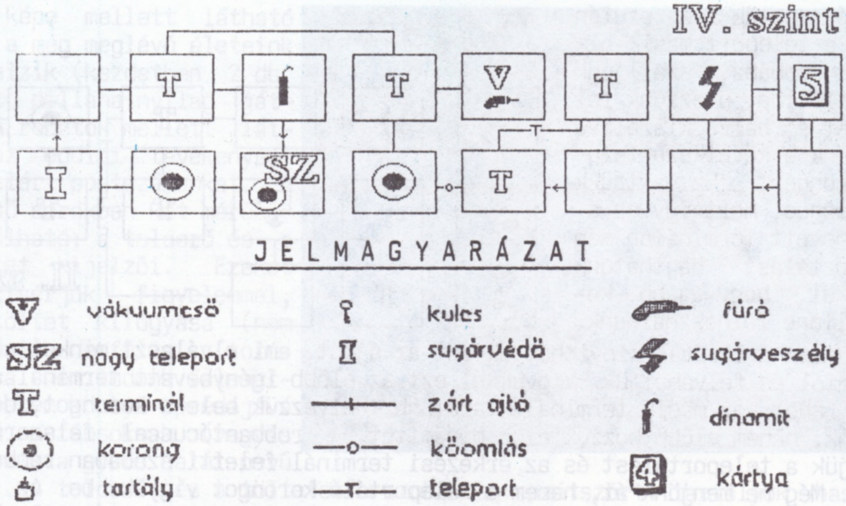


A III. szintre történő belépés után menjünk el a kulcsért, amellyel kinyithatjuk a lenti régiók egyik szobájában lévő ajtót (ld. térkép). Az ajtó mögött lévő szobában vegyük magunkhoz a sugárvédőt. Ennek segítségével a sugárveszéllyel telített szobában (bal felső) a sugárzás megszűnik és felvehetjük a teleportáló korongot. A korongot vigyük le a legalsó szinten lévő terminálhoz, ahonnan teleportálva a jobb felső szobában találjuk magunkat. Már mosolyog is ránk a harmadik tartály, vegyük fel szépen. Menjünk le a vákuumcsőhöz (ne teleportálással!) és helyezzük el benne a tartályt. Felvillan az ismerős szoba képe és lőn csoda, MÁR HÁROM BUTÉLIA VAN A POLCON.

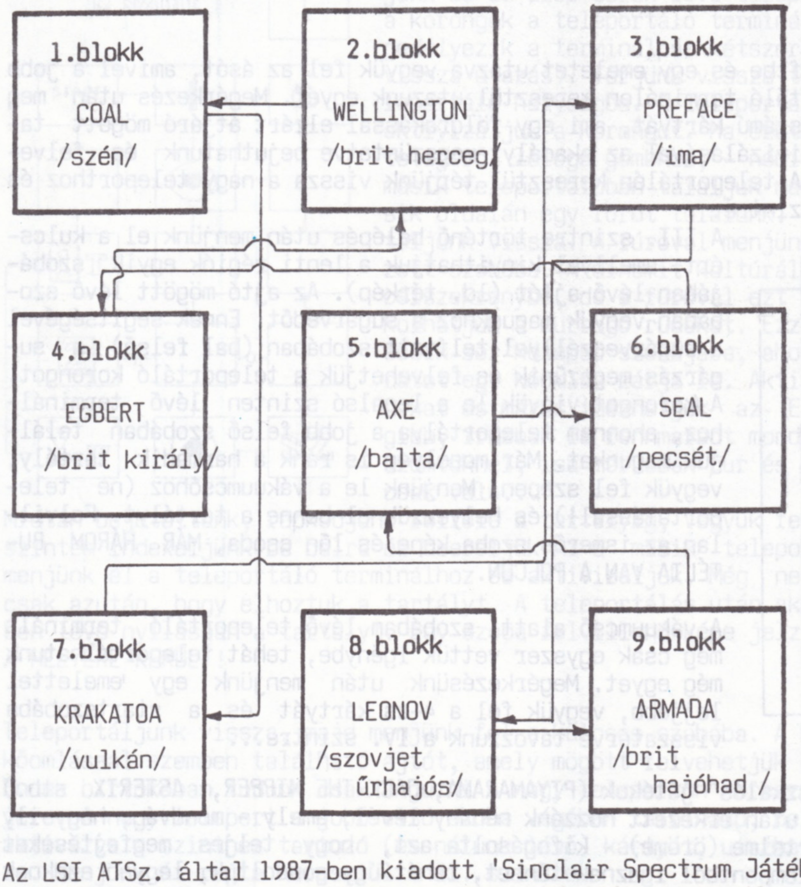
A vákuumcső alatti szobában lévő teleportáló terminált még csak egyszer vettük igénybe, tehát teleportálhatunk még egyet. Megérkezésünk után menjünk egy emelettel lejjebb, vegyük fel a 4-es kártyát és a startszobába visszatérve távozzunk a IV. szintre...

Az eddig megjelent "mászkálós" játékok (PIYAMARAMA, JACK THE NIPPER, ASTERIX stb.) leírásának megjelenése után érkezett hozzánk néhány levél, amely - mondván, hogy ily módon elvész a játék értelme (öröme) - kifogásolta azt, hogy teljes megfejtéseket közlünk. Ez bizonyos szempontból igaznak tűnhet, tehát úgy gondoltuk, legyen ezeknek az uraknak is egy jó napjuk: ennél a pontnál Rémusz bácsi ifjú kollégája befejezi az Equinox programról szóló mesét. A IV. szint térképét még megmutatjuk, a többi pedig legyen a tisztelt levélírók dolga. Jó szórakozást!

2.2 EQUINOX



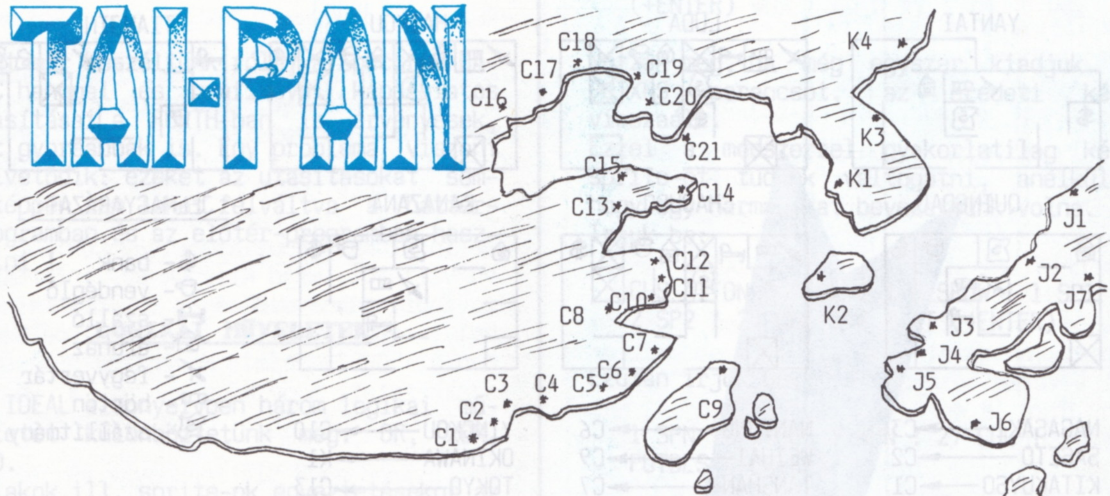
STARION 2.zóna



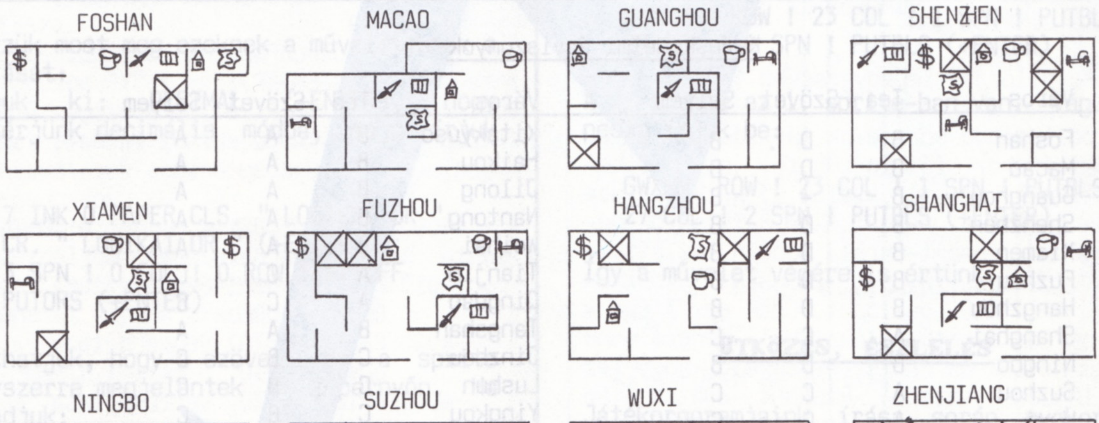
Blokk	Probléma	Szállítmány
1:	A herceg piszkos lesz Waterloo-nál	WELLINGTON
2:	Anglia tejhatalmú uralkodója	EGBERT
3:	Nagy robajjal kitör Indonéziában	KRAKATOA
4:	Érvénybe lép az USA függ.nyilatkozata	SEAL
5:	Drake szétveri a spanyolokat	ARMADA
6:	Mary Liz gyerekkosarában találta	AXE
7:	Mi hajtja James Steam motorját	COAL
8:	Így kezdődik az angol földbirt.könyv	PREFACE
9:	Űrséta	LEONOV

Az LSI ATSz. által 1987-ben kiadott 'Sinclair Spectrum Játék és Program' c. könyvsorozat III.részében a STARION 1.zóna megfejtése ismeretlen maradt. Íme a megoldás (amely nem is volt olyan bonyolult): RED BREAST /vörösbegy/. A most ismertetett 2. zóna kulcsszava: SPACE WALK /űrséta/.

TAI-PAN



C1 = FOSHAN	C9 = NINGBO	C17 = KITAKYUSO	K4 = JINZHOU
C2 = MACAO	C10 = SUZHOU	C18 = HAIKOU	J1 = LUSHUN
C3 = GUANGHOU	C11 = WUXI	C19 = JILONG	J2 = YINGKOU
C4 = SHENZHEN	C12 = ZHENJIANG	C20 = NANTONG	J3 = CHAJUDO
C5 = XIAMEN	C13 = YANTAI	C21 = WEIHAI	J4 = OKINAWA
C6 = FUZHOU	C14 = LUDA	K1 = TIANJIN	J5 = TOKYO
C7 = HANGZHOU	C15 = NAGASAKI	K2 = QUINGDAO	J6 = KANAZANA
C8 = SHANGHAI	C16 = SASETO	K3 = TANGSHAN	J7 = KWANGSU



Barbarian

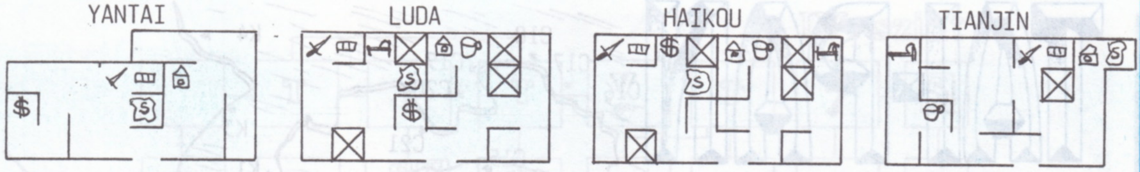


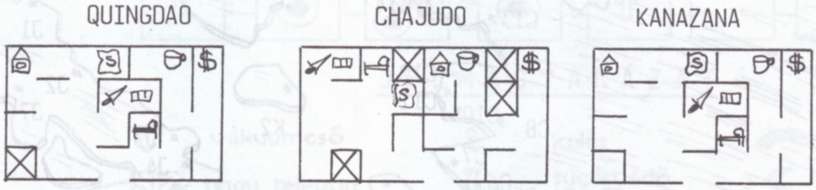
A program file-térképe a következő: 6912/20000/20536/6916/1704
A sérthetelenséghez töltsük be a LOADER-t: MERGE"". Írjuk be a következő parancsokat:

```
POKE 23800,62: POKE 23801,68: POKE 23802,50:
POKE 23803,179: POKE 23804,153 (ENTER) RUN (ENTER)
```

és indítsuk el a magnót.

2. JÁTÉK, POKE, TÉRKÉP





JELMAGYARÁZAT:

- $ - bank
- vendéglő
- szálló
- áruház
- fegyvertár
- börtön
- szállítmány

NAGASAKI → C3
SASETO → C2
KITAKUYSO → C1
JILONG → C5
NANTONG → C6
WEIHAI → C9
TANSHAN → C7
LUSHUN → C11
YINGKOU → C10
OKINAWA → K1
TOKYO → C13

A szállítmányok árkategóriái:
(ezer dollár)

	TEA	SZÖVET	SELYEM
A	23-25	64-66	36-39
B	25-29	71-74	39-42
C	29-31	78-80	42-45
D	-----	85-87	-----

Kereskedelmi árviszonyok

Város	Tea	Szövet	Selyem
Foshan	B	D	B
Macao	B	D	B
Guanghou	B	-	B
Shenzhen	B	D	B
Xiamen	B	D	B
Fuzhou	B	D	B
Hangzhou	B	D	B
Shanghai	A	C	C
Ningbo	B	D	B
Suzhou	A	C	C
Wuxi	A	C	C
Zhenjiang	A	C	C
Yuntai	A	C	C
Luda	A	C	C
Nagasaki	B	A	A
Saseto	B	A	A

Város	Tea	Szövet	Selyem
Kitakyuso	B	A	A
Haikou	B	A	A
Jilong	B	A	A
Nantong	B	A	A
Weihai	B	A	A
Tianjin	A	C	C
Qingdao	A	C	C
Tangshan	B	A	A
Jinzhou	C	B	C
Lushun	C	B	C
Yingkou	C	B	C
Chajudo	C	B	C
Okinawa	C	B	C
Tokyo	C	B	C
Kanazana	C	B	C
Kwangsu	C	B	C

The Goonies

File-térképe: 164/6912/20000/20536/6916/1704
A végtelen élethez töltsük be a LOADER-t MERGE"". Majd írjuk be a következő parancsokat: 20 REM (ENTER) POKE 23757,96 (ENTER) majd a következő BASIC programot:

```
30 FOR i= 23841 TO 23850: READ a: POKE i,a: NEXT i
40 DATA 175,50,129,130,61,55,229,195,86,5
```

Futtassuk a programot (RUN+ENTER) és indítsuk el a magnót.

FORTH/BASIC SZAVAK

Mint már beszéltünk róla, a Spectrum BASIC hanggal és grafikával kapcsolatos utasításai a FORTH-ban is érvényesek, sőt gyorsabbak is. Egy probléma viszont felvetődik: ezeket az utasításokat semmiképpen sem lehet felváltva a háttér-programban és az előtér-programban használni !

LOGIKAI MŰVELETEK

Az IDEAL alapnyelvben három logikai műveletet különböztetünk meg: OR, XOR, AND.
Ablakok ill. sprite-ok egyeztetésekor az azonos pozícióban elhelyezkedő képpontok állapota alapján más és más végeredményünk születik. Könnyítésképpen tekintsük át a logikai műveletek igazságtábláit:

```
1 OR 1 = 1    1 XOR 1 = 0    1 AND 1 = 1
0 OR 1 = 1    0 XOR 1 = 1    0 AND 1 = 0
1 OR 0 = 1    1 XOR 0 = 1    1 AND 0 = 0
0 OR 0 = 0    0 XOR 0 = 0    0 AND 0 = 0
```

Nézzük most meg ezeknek a műveleteknek a hatását:
Adjuk ki: DECIMAL (+ENTER), hogy áttérjünk decimális módba, majd írjuk be:

```
7 INK 0 PAPER CLS. " LOGIKAI OR "
CR. " LOGIKAI OR " (+ENTER)
3 SPN ! 0 COL ! 0 ROW ! ATTOFF
PUTORS (+ENTER)
```

Láthatjuk, hogy a szöveg, és a sprite egyszerre megjelentek a képernyőn. Ha kiadjuk:

```
CLS. " LOGIKAI AND " CR. " LOGIKAI
AND " PUTNDS (+ENTER)
```

Ennek következtében csak ott marad képpont-adat, ahol az adatok megegyeznek. Ez a módszer a leghatásosabban az ütközések észlelésére használható fel. De ez még mind semmi, a három közül a legelőnyösebb a logikai XOR, egyébként a Sinclair BASIC OVER utasítása is így működik.

```
CLS. " XOR " CR. " XOR " PUTXRS
(+ENTER)
```

Nézzük meg, ha még egyszer kiadjuk a PUTXRS parancsot, az eredeti kép visszaáll.
Ezzel a módszerrel gyakorlatilag két sprite-ot tudunk váltogatni, anélkül, hogy egy harmadikat bevezettünk volna.
Írjuk be:

```
CLS ATTON 0 SCOL 0 ! 0 SROW ! 1 SP1 !
2 SP2 ! 23 COL ! 6 0 AT (+ENTER)
```

Ezután írjuk be:

```
1 SPN ! PUTBLS 2 SPN ! 27 COL !
PUTBLS (+ENTER)
```

Vagyis a művelet kezdete előtt rendelkezésünkre áll két db. sprite.

```
GWXRM 2 ROW ! 23 COL ! 1 SPN !
PUTBLS 27 COL ! 2 SPN ! PUTBLS
(+ENTER)
```

Végrehajtottunk az 1. és a 2. sprite között egy XOR műveletet.

```
PWXRM 4 ROW ! 23 COL ! 1 SPN ! PUTBLS
27 COL ! 2 SPN ! PUTBLS (+ENTER)
```

A 2. sprite az 1. sprite-ban van. Végül pedig írjuk be:

```
GWXRM ROW ! 23 COL ! 1 SPN ! PUTBLS
27 COL ! 2 SPN ! PUTBLS (+ENTER)
```

Így a művelet végére is értünk.

ÜTKÖZÉS, ÉSZLELÉS

Játékprogramjaink írása során gyakori feladat az ütközési jelenségek megtervezése, annak észlelése, az ütközést követően új szubrutinra való áttérés stb. Ebben a dologban nagy segítségünkre vannak a SCANV és a SCANM utasítások.
A SCANV segítségével egy képernyő-karakter pozíció vizsgálatára nyílik lehetőségünk. Az adott karaktert a COL és a ROW változók határozzák meg. Ha az adott karakter belsejében valamelyik képpont egyes értékű, úgy a verem tetején a program 1-t helyez el, máskülönben 0-t.

```
CLS VLIST 0 VARIABLE CNT (+ENTER)
:GO 0 CNT ! 24 0 DO I ROW ! 32 0
(+ENTER)
DO I COL ! SCANV IF 1 CNT - (+ENTER)
ENDIF LOOP LOOP CNT - ; (+ENTER)
```

A GO most egyszerűen számolja a képernyőn lévő karaktereket. Ha beírjuk: GO (+ENTER), kb. 250-et fog a program kiírni.
Hasonló feladatot lát el a SCANM, amely az SPN-nek megfelelő sprite-ot vizsgálja meg, van-e benne adat. Igen esetén egyes, nem esetén zérus értéket helyez el a verem tetején.
Egy ütközés észleléséhez szükséges egy néma sprite, melybe bekerül a sprite, majd AND kapcsolatba lép az ablakkal, és itt kerül végrehajtásra a SCANM is.
Ha észleljük az ütközést, hasznos dolog meghatározni, hogy a sprite mivé váljon az ütközés hatására.

SCROLLOZÓ TÁJAK

A leglényegesebb, hogy soha ne scrollozzunk többet, mint amennyi szükséges. Pl. hegyi táj esetén csak a legfelső karaktersorban történnek variációk, így ezt kell eltárolnunk.

A scrollozás legegyszerűbb módja, egy oszlop feláldozása a képernyőből a scrollozandó sprite-ok tárolásához, átalakításához. Tételezzük fel a következőt: 4-5 karakter széles, 3 karakter magas sprite-ot mozgatunk a 8-10 sorok mezőjében képpont-scroll módban. Az attributumok vízszintesen azonosak. Áldozzuk fel most a 0. oszlopot. Csak egy 3 sor x 1 oszlop méretű ablakot kell generálnunk azonos INK és PAPER értékkel, hogy az állandóan változó képpont adatok ne látsszanak. Ehhez célszerű a SETAV utasítást felhasználni. Tegyük ki a képernyő aktív részére a sprite 31 oszlopát a PWBLS felhasználásával. Balra scroll esetén a sprite-tól jobbra következő oszlopot kell betölteni az un. néma sprite-ba. Így gyakorlatilag a táj össze lesz tekercselve, amíg a kép nem lép tovább 8 képpontnyit. Ezután a megfelelő, következő oszlopot kell betölteni a néma sprite-ba és így tovább. A scroll egyébként hatékonyan alkalmazható kettes megszakítás alatt.

PROGRAMOZHATÓ SPRITE-OK

A háttér-programok leggyakoribb felhasználási lehetősége, sprite-ok automatikus mozgásba való beállítása.

```
0 VARIABLE CONTROL CONTROL U. (+ENTER)
```

Az itt látható parancssor segítségével definiálunk, kinullázunk egy CONTROL nevű változót, majd kiírjuk az információ továbbításához szükséges memória kisebb helyiértékű byte-ját. Legyen ez példánkban az 50000. cím. Használjuk a CASE szerkezetet, hogy kiválasszuk a végrehajtandó szót. Írjuk be a következő definíciót:

```
:SELECT CASE 1 OF . " SZO1 " ENDOF
2 OF . " SZO2 " (+ENTER)
ENDOF 3 OF . " SZO3 " ENDOF 4 OF .
" SZO4 " ENDOF (+ENTER)
ENDCASE CR ; (+ENTER)
```

A CONTROL változó értékének (1-4) megfelelő szó (SZO) kerül végrehajtásra.
Most írjuk be a következőt:

```
:RUN 4000 GOTO BEGIN CONTROL@
(+ENTER)
DUP SELECT DUP IF RETUSR ENDIF 0=
UNTIL ; (+ENTER)
```

Programunk a 4000. sortól futhat, formája lehet pl. a következő:

```
4000 REM az 5000-es cím és a CONTROL
     címe felcserélése
4010 PRINT " SZO1 vegrehajtasa ":
     GO SUB 5000
4020 PRINT " SZO2 vegrehajtasa ":
     GO SUB 5010
4030 PRINT " SZO3 vegrehajtasa ":
     GO SUB 5020
4040 PRINT " SZO4 vegrehajtasa ":
     GO SUB 5030
4050 PRINT "VEGE": POKE 50000,0:
     PRINT USR 30006
5000 POKE 50000,1: RANDOMIZE USR
     30000: RETURN
5010 POKE 50000,2: RANDOMIZE USR
     30000: RETURN
5020 POKE 50000,3: RANDOMIZE USR
     30000: RETURN
5030 POKE 50000,4: RANDOMIZE USR
     30000: RETURN
```

Végül a gép a PRINT USR 30006 utasítást hajtja végre.
Használjuk a RETUSR utasítást, máskülönben a BASIC verem tartalma tönkremegy. Ha ez mégis megtörténne, célszerű kiadni a PROG parancsot, amire áttérünk a BASIC rendszerhez, ezután pedig a meleg-start segítségével lépjünk vissza a FORTH-ba (PRINT USR 24836).

PROGRAM ÍRÁSA

A programírás javaslatunk szerint 5 jól elkülönített fázisból kell hogy álljon, röviden tekintsük át ezeket:

1/ FORTH forrás

A forrás egyenként 512 byte hosszú képernyőkre van bontva. Mindegyik külön-külön tölthető, menthető, szerkeszthető, a kívánt sorrendben. Az elmentett képernyők természetesen más képernyőbe áttölthetők. Ha pl. a 6-11-ig terjedő screen-eket el akarjuk menteni, ehhez először is át kell térni a BASIC-hez a PROG segítségével, majd SAVE "név" CODE 52224,3072, mivel 6x512=3072.
A White Lightning-be ismét a meleg start-tal (PRINT USR 24836) léphetünk vissza.
Forrás betöltéséhez a "LOAD SOURCE Y/N" kérdésre "Y"-t kell gépelnünk, vagy a PROG segítségével áttérhetünk a BASIC-hez, és LOAD "név" CODE. A start, hossz paraméterek eltérő adatok esetén külön megadhatók.
FONTOS!!! Meleg start esetén soha ne használjunk RANDOMIZE USR 24836 utasítást, mert "OUT OF SCREEN ERROR" hibaüzenetet kapunk eredményül.

2/ BASIC forrás

Használata előtt egy RESERVE segítségével helyet kell biztosítani a BASIC program részére. 2 Kbyte helyfoglaláshoz írjuk be: DECIMAL 2048 RESERVE. Második RESERVE kiadása (pl. kevés helyfoglalás esetén) nem javasolt, helyette célszerű elmenteni az összes forrást, az előző definíciókat töröljük FORGET-tel, hajtsunk végre hidegindítást, s ezután már jöhet a második RESERVE. A forrás kimentéséhez térjünk át a PROG segítségével BASIC-be, majd SAVE "név". Visszatérés a melegindítással lehetséges.

3/ Sprite-ok és MERGE-elésük

A WL sprite-jai ki/betölthetők a WL ból/ba, de más sprite tervező software-be már nem, a tömb információk hiánya miatt. A sprite-ok kimentése a következőképpen lehetséges:
Keressük meg a sprite-ok kezdőcímét, SPST @ U., majd hosszát, SPND @ SPST@ - 1 + U., ezután térjünk vissza a BASIC-hez, PROG, végezzük el az elmentést, SAVE "név" CODE start, hossz, és térjünk vissza a WL-ba, PRINT USR 24836.

Először is jegyezzük meg a második blokk SPST és SPND értékét. Töltsük be a WL-t, a 'LOAD SPRITES Y/N' kérdésre válaszoljunk 'Y'-t, és töltsük be az első sprite-blokkot. Töltsük be a forrást, relokáljuk lefelé az első blokkot a második méretével. Pld. a 2. blokk SPST értéke 60000, SPND értéke 65280, akkor adjuk ki: DECIMAL 60000 65280 -SPST +U. (+ENTER), majd DECIMAL 60000 65280 -MLEN ! RELOCATE (+ENTER). A MLEN tárolja a relokációs hosszt. Negatív esetben lefelé, pozitív esetben felfelé történik a relokáció. A 2. blokk betöltése előtt ki kell számítani ill. meg kell jegyezni az új SPST-t és az SPND-t:

```
SPST U. SPND U. 65280 60000 - + DUP
SPND U. (+ENTER)
```

Ha előtte minden rendben volt, az új SPND megegyezik a relokáció előttivel. Ezután végezzük el a betöltést: PROG (+ENTER), majd LOAD "" CODE, végül pedig térjünk vissza melegindítással: PRINT USR 24836.

4/ A FORTH és bővítése

A FORTH legnagyobb előnye, hogy szókészlete bővíthető. A bővített szótár hoszszának ismeretében BASIC-ből mentsük ki a bővített szótárt: SAVE "FORTH" CODE 24832, hossz, és térjünk vissza melegindítással. A kibővített FORTH használatához először is töltsük be a WL-t, majd PROG, vagyis térjünk vissza a BASIC-hez és töltsük be az előzőleg kimentett FORTH-t a régi helyére. Végül hidegindítást hajtsunk végre a PRINT USR 24832 utasítás segítségével.

5/ A megszerkesztett és befejezett program

A legvégén ZAP utasítást, ha használunk hátter-programot, akkor ZAPINT utasítást kell kiadni. Kiíródik a program hossza, és átadódik a vezérlés a BASIC-hez. A kész program elmentése a SAVE "név" CODE 24832, hossz utasítással lehetséges. Elmentés után a PRINT USR 24832 utasítással adjunk melegindítást. Megjegyezzük, hogy White Lightning-gel készült programok, csak ilyen formában forgalmazhatók.

UTÓSZÓ

Élérkeztünk tehát a WHITE LIGHTNING programrendszer útmutatójának végére. Bevalljuk őszintén, kezdetben nehéz volt felmérnünk, milyen részletességgel érdemes taglalnunk az egyes fejezetpontokat. Úgy érezzük ez a leírás elegendő ahhoz, hogy azok, akik szeretnének mélyebben belemászni a White Lightning lelki világába, azok saját maguk továbbfejlesszék mindazt, amit itt olvastak, és tanultak. Elismerjük, hogy ez a sorozat mintapéldában igen szegényes volt, de erről a programrendszerről önálló 200 oldalas könyvet is lehetne írni. Akik

FORTH forrás screen-cím táblázat

Screen száma	1	2	3	4	5
-cím	49664	50176	50688	51200	51712
	6	7	8	9	10
	52224	52736	53248	53760	54272
	11	12	13	14	15
	54784	55296	55808	56320	56832
	16	17	18	19	20
	57344	57856	58368	58880	59392
	21	22			
	59904	60416			

mégis úgy gondolják, hogy az itt közölt információk elegendőek a White Lightning meghódításához, azok vágjanak bele, érdemes. Nagyon sok jelenleg forgalomban lévő játékprogram konkrét mechanizmusának a megtervezése a White Lightning segítségével készült, elég, ha csak a magyar szerzők által elkészített és a DOMARK cég által forgalmazott EUREKA-t említjük.
Búcsúzik tehát a White Lightning, de talán nem véglegesen. Amennyiben egy-egy témakörről mégis kevés, vagy nem teljesen érthető információt közöltünk volna, úgy arra (azokra) a későbbiekben még valószínűleg visszatérünk.

TARTALOMJEGYZÉK:

A PSION Software által 1983-ban útjára bocsájtott VU-3D jól kivitelezett 3D grafikai tervező program. Legfőbb jellegzetessége a menücentrikus kezelési rendszer. A programmal lehetőség nyílik a 3D test könnyű megszerkesztésére. Ez úgy végezhető el, hogy a test 3D z-koordináta szerinti x-y szeleteit sorban elkészítjük. A program az összetartozó csúcspontokat automatikusan összeköti, s bár ez egy kicsit megköti a kezünket, ugyanakkor előnyös is. A megszerkesztett test képét előhívhatjuk az elforgatott 3D koordináta rendszerbe. A program legfőbb jellegzetességét ettől a pillanattól kezdve használhatjuk ki. A forgatási láncon túlmenően eltüntethetjük a nem látható éleket (hidden-line), vagy megvilágíthatjuk és beárnyékolhatjuk különböző irányból a testet (shade). A program forgatási algoritmusa nem a legszerencsésebb, így még egyszerűbb alakzatok esetén is igen szaggatott a forgatás és ez egy kicsit illúzióromboló. Szintén nem válik a program előnyére a kész alakzat kötött módosítási lehetősége, amelyet már csak a kijelölt z szeletekben végezhetünk el, utólag újabb szeleteket nem generálhatunk. Természetesen ettől függetlenül a VU-3D a maga idejében egyedülálló képességeivel magáért beszélt.

A program felépítése a kazettán a következő:

pr:	VU3D	(35 byte)
pr:	vu3D	(6754 byte)
by:	s	(6912 byte)
by:	c	(32183 byte)
by:	EXAMPLE	(12330 byte)

A kazettán találhatunk egy demo programot is EXAMPLE néven, amely csak a főprogram betöltése után hívható be (ld. később).

A VU-3D egyszerűen betölthető a:
LOAD "VU3D" vagy a LOAD""
parancsokkal.

A betöltés után bejelentkezik egy menü:

Do you wish to:
1/ Create a new figure
(Új kép készítése)
2/ Load a data file (Kazettás adatállomány betöltése)

Ha először használjuk a VU-3D-t, célszerű az 1. opciót választani, a 2. opcióval előzetesen kimentett adatainkat tölthetjük be, így pl. a demo-t is.

A VU-3D kezelését tekintsük meg a menüfunkciók segítségével:

C R E A T E

A CREATE menübe az 1. opció választásakor kerülünk.

CREATE Open Close Figure
Magnify Quit Next z Reduce
←,→,↓,↑ to move figure

Ez a funkció lehetőséget biztosít a felhasználónak, hogy egyszerű parancs-lépésekkel elkészítsen egy új 3D tárgyat. A képernyő felső három sorában jelennek meg a CREATE parancsok. Az alsó sor az adatmező. A parancsok egyszerűen meghívhatók, ha megnyomjuk első betűjük, vagy szimbólumuk billentyűjét a billentyűzeten. A CREATE parancsok egymás után végrehajthatók, a testen un. 3D szeletekre bontva. A 3D tárgy minden szeletének, metszetének van egy aktuális z koordinátája. Minden z koordináta a 3D tárgynak csak egy bizonyos kétdimenziós szeletét határozza meg, mintha pl. egy bonyolult tárgyat késsel több helyen elszeletelnénk. Egy adott szeletben a tárgy képe x-y koordinátarendszerben jelenik meg. Az x-y koordináta rendszer 0 pontja a képernyő bal-alsó sarkában látható.

Mielőtt elkezdjük az ábra készítését, célszerű a tervezendő tárgyunkról papíron sematikus vázlatot készíteni, melyen jelöljük a meghatározott szeleteket, és az egyes pontok koordináta értékeit. Ez egy adott szeletben megkönnyíti a tervezést a képernyőn, hiszen az adatmezőben csak az x és y koordináta-értékeket kell figyelnünk, másrészt figyelembe kell venni azt is, hogy koordináta rendszerünk z irányban 0-tól terjed 254-ig. Ha egy rugbylabdát szertnénk felrajzolni, annak egyik vége pl. a z=0, míg másik vége a z=254 szeletben lesz, és ezekben a szeletekben (a rugby-labda két végén) csak egy-egy pont látható.

A következőkben tekintsük át a CREATE menüből elérhető parancsokat.

O P E N

Ez a tárgy elkészítésének kezdete, a rajzolási lehetőség engedélyezése. Eléréséhez egyszerűen nyomjuk meg az 'O' billentyűt. A parancs lehetőséget biztosít az adott szeletben egy kép megrajzolására.

A parancs-sorban a következő jelenik meg:

OPEN Start Line Delete End
←,→,↓,↑ to move cursor
CAPS SHIFT-slow cursor

Az adatmezőben pedig a következőt látjuk:
CURSOR AT x = 120 Y = 072 Z = 000

A képernyőn egy kurzor jelenik meg.
Első lépésként próbáljuk meg a kurzort mozgatni a kurzor-vezérlő billentyűkkel. Ha eközben megnyomjuk a CAPS SHIFT-et is, a kurzor lassabb lesz, így a pontos pozícionálás nem jelent problémát. Ceruzánkkal még csak a papír felett mozgunk. Ha vonalat szeretnénk húzni, a ceruza hegyét le kell tenni a papírra, pontosabban meg kell adni a vonal kiinduló pozíciójának helyzetét.
Ez a "Start" paranccsal érhető el, nyomjuk meg az 'S'-t. A következő feladat, hogy kurzorunkat mozgassuk át abba a pozícióba, ameddig az egyenest szeretnénk meghúzni, majd 'L' (Line) megnyomására megjelenik az egyenes a két kijelölt pont között. Ha ezt a rajzot folytatjuk tovább, újabb 'S' megnyomására nincs szükségünk, a kiinduló pont mindig a legutoljára berajzolt egyenes végpontja lesz.
Így mozgassuk most ismét át a kurzort egy új pozícióba, és nyomjuk meg 'L'-t.
A program szabályos kör definiálására nem képes, ellenben sokszög-poligon (pl. a rugby-labda esetében) rajzolható, az előbb megismert módon, sok-sok kis egyenes kapcsolatából.
A 'D' (Delete) a legutoljára megrajzolt vonalat fogja kitörölni, ha ismételten megnyomjuk akkor az azelőttit, és így tovább, végül visszaadja a CREATE menüt.
Az 'E' (End) paranccsal is visszaléphetünk a CREATE menübe, de előbb az adatsorban megjelenik egy üzenet: "MAGNIFICATION = x.xx" amely az éppen aktuális nagyítás értékét jelzi. Alapesetben ez normális, vagyis értéke: 1.00.
Az ábra felrajzolása után 'O' (Open) az előző szeletben újabb ábra megrajzolását teszi lehetővé, függetlenül a másiktól, így térben egymás mellett akár több testet is felrajzolhatunk.

Megjegyzés: Csak zárt síkidom rajzolására van lehetőség (szabályos vagy szabálytalan sokszögek), így amíg zárt vonallal nem térünk vissza a kiinduló pozícióba, a program nem enged visszatérést a CREATE menübe.
Ha egymás mellett egy adott szeletben több síkidomot is rajzolunk, azok között nem lehet semmiféle átfedés, mert a program nem fogadja el.

F I G U R E

Mint az előbb említettük, egy z szeleten belül lehetőség van több azonos vagy különböző, de egymástól független kép elhelyezésére. Az egyes képek egymástól függetlenül nagyíthatók (MAGNIFY), kicsinyíthetők (REDUCE), illetve mozgathatók (kurzor billentyűk) a képernyőn. Az "F" (FIGURE) parancs több megszerkesztett kép között enged meg átváltást. Az egyes műveleteket mindig csak a kiválasztott képpel végezhetjük el.
Megjegyzés: Ha csak egy képünk van az adott z szeletben, a FIGURE parancs hatástalan.

M A G N I F Y

A kiválasztott kép az "M" (MAGNIFY) paranccsal nagyítható a képernyőn. A nagyítás mindenkori értéke az alsó adatmezőben látható.

R E D U C E

A MAGNIFY párja, segítségével a kiválasztott kép kicsinyíthető a képernyőn. A kicsinyítés mértéke szintén nyomon követhető az alsó sorban.

←→↑↓ MOZGATÁS

A kiválasztott kép balra/jobbra/fel/lefelé történő mozgatását segíti elő.

N E X T z

Ha a kiinduló szeletben (z=000) elvégeztük a kép felrajzolását, az "N" (NEXT z) paranccsal a z koordináta léptéket változtathatjuk, növekvő irányban. Minden egyes 'N' megnyomására 1-gyel tovább lép a z értéke. Az intervallum 0-254-ig terjed, és sajnos csak növekvő irányban léptethető. Gondoljunk ismét a rugby példára. A z=000 szeletben elkészítjük a sokszög-poligont, majd lekicsinyítjük egy pontra. z=127-ig folyamatosan nagyítjuk (először gyorsabban, később lassabban), 127-től pedig ismét kicsinyítjük, végül z=254-nél újra csak egy pontot látunk.
Megjegyzés: Egy adott z szeletben mindig csak azt jegyzi meg a program, ami éppen akkor a z továbblépése pillanatában a képernyőn látható.

C L O S E

Ha elkészítettük a 3D tárgy összes szeletét (ehhez mindenképpen z változásnak kellett történnie), akkor a 'C' (CLOSE) paranccsal lezárhatjuk a térbeli alakza-

tot. Ez nem jelenti azt, hogy felette (pl. már z=120-nál lezártuk) újabb kép nem rajzolható fel.
'C' megnyomására a

DO YOU WANT TO TOP OPEN Y/N ?

kérdésre 'N'-nel kell válaszolnunk, ha az adott képet szeretnénk lezárni.

Q U I T

A "Q" (QUIT) parancs segítségével meghívhatjuk a főmenüt:

M A I N M E N U (FŐMENÜ)

1. Modify a figure
2. Abandon
3. Load a data file
4. Create a new figure
5. Save a data file
6. Display
7. Change Colours

A főmenü egyes funkcióinak eléréséhez az előttük levő számot kell begépelnünk, majd meg kell nyomni az ENTER-t.

M o d i f y a f i g u r e

Ezt a funkciót akkor hívjuk meg, ha elkészített tárgyunk képe nagyjából megfelelő, de kisebb módosításokra még szükség van. A CREATE menühöz hasonló, de itt nem kell végigmenni 000-254-ig az összes z szeleten. Az 'N' (NEXT z) parancsra sorban csak azokat a z szeleteket adja elő a program, amelyeken a meghatározott képeket elkészítettük. A módosítás a CREATE menüben leírtak szerint elvégezhető. Bármely szelet javítása, módosítása után, ill. az 'N' (NEXT z) utolsó szeletet követő megnyomása után ismét megjelenik a főmenü.

A b a n d o n

A memóriából törli az éppen bent levő adatállományt. Ezt a funkciót akkor szoktuk meghívni, ha új képet szeretnénk rajzolni, vagy egy új file-t akarunk betölteni a kazettás magnetofonról.

L o a d a d a t a f i l e

Ennek az opciónak a segítségével behívhatunk a kazettás magnetofonról egy előzetesen kimentett adatállományt. Betöltés előtt a gép kéri tőlünk a file nevét.
Megjegyzés: a demo kazetta B oldalán található egy bemutató file EXAMPLE (másolt magyar forgalomban "pohar") néven, amely szemléletesen bemutatja a program lehetőségeit.

C r e a t e a n e w f i g u r e

Ismét a CREATE menübe lépünk be, s innentől az előzőleg leírtak szerint kell eljárni.

S a v e a d a t a f i l e

A memóriában levő adatállományt kód formájában kazettás magnetofonra menti. Kimentés előtt a program bekéri a file nevét.

D I S P L A Y

A 6. opció meghívásakor egy újabb 3 soros menüt kapunk, a DISPLAY menüt:

```
DISPLAY  Magnify Reduce Picture
 Near   Far   CAPS SHFT-slow
 Quit  ←-↻  →-↺  ↑-↑ ↓-↓
```

Ezen kívül megjelenik a definiált 3D tárgy a 3D képében, valamint az alsó sorban az adatmező, amely a nagyítás, a forgatás és a z koordináta adatait tartalmazza. A tárgyat "üvegszerűnek" látjuk, vagyis a "nem látható" élek is látszanak.

F a r

Az 'F' parancs hatására távolodik a tárgy a néző szemszögből. CAPS SHIFT + F segítségével a folyamat lassítható.

N e a r

Az előző ellentét-párja, az 'N' megnyomására a tárgy közelebb kerül hozzánk. CAPS SHIFT + N segítségével a folyamat lassítható.

Irány nyilak

A tárgy vízszintes és függőleges perspektívában történő forgatását segítik elő.

M a g n i f y

Az 'M' paranccsal a képernyőterület nagyítható. Figyelem! Nem azonos a Near paranccsal, itt csak nagyításról van szó, de a perspektíva nem változik.

R e d u c e

Az 'R' parancs az 'M' ellentét-párja. Nem azonos a Far paranccsal, mivel perspektívikus változást nem idéz elő.

Q u i t

A 'Q' parancs segítségével visszatérhetünk a főmenübe.

P I C T U R E

A 'P' megnyomására beléphetünk a PICTURE menübe:

```
PICTURE  Shade   Hiddenline
 Print  Colour  Keep   Quit
```

4. VU-3D

Hiddenline
Ez egy igen jó szolgáltatása a programnak. Kis gondolkodási idő után eltűnnek a testről a mögöttes élek, s ezután már nem "üvegszerű" ábrát látunk, hanem tömör nézeti képet. Több egymás melletti 3D test esetén az egyes testek takaró hatása is szépen megjeleníthető.

Shade
Ez egy másik út a 3D látószög szép szemléltetésére. Feltételezünk egy egyszerű fényforrást, ami megvilágítja a tárgyunkat, és árnyékot vet arra. Megválasztására megkapjuk azt a kérdést, hogy a fényforrásból (LIGHT SOURCE) a fény milyen szögből érkezzen:

- ABOVE (Felülről)
- CENTRE (Középről)
- BELOW (Alulról)

Ezt követően azt kell megadni, hogy milyen legyen a horizontális megvilágítási irány:

- LEFT (Balról)
- CENTRE (Középről)
- RIGHT (Jobbról)

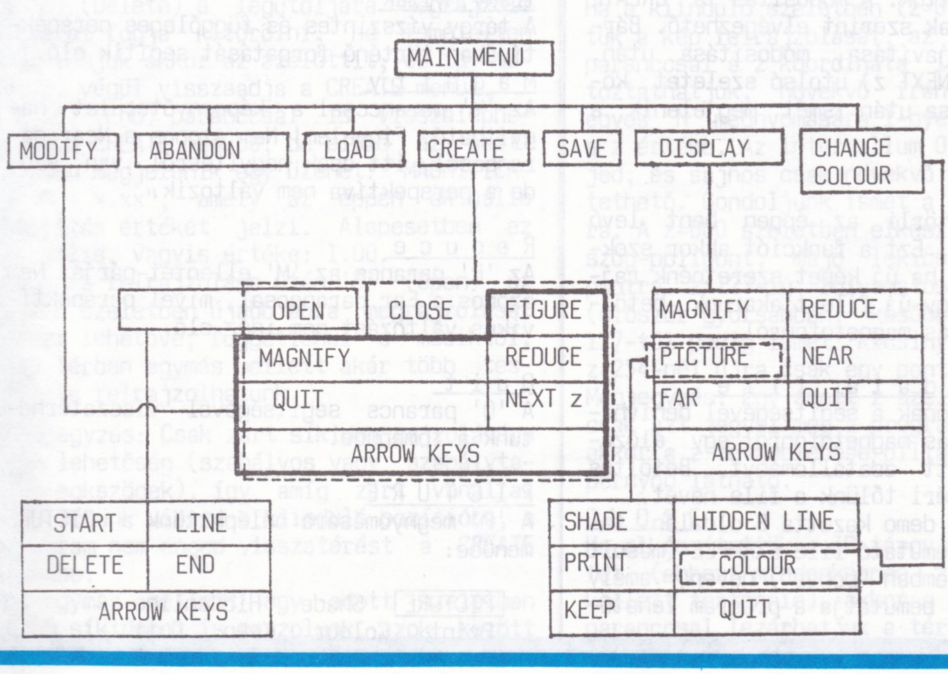
Végül a SHADING üzenet jelzi a feladat végrehajtását, majd visszakapjuk a PICTURE menüt.

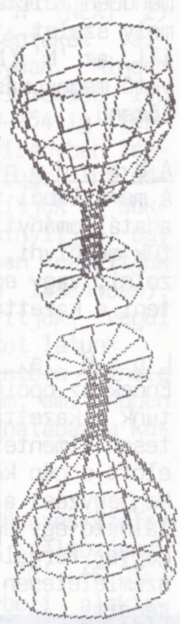
Colour
A Colour parancs segítségével egyes képernyőterületek átszínezhetők. Ha megnyomjuk a "C" billentyűt, először a keret (BORDER) színét kéri be a gép, majd sorban az üzenetmező (BANNER), háttér (PAPER) és kiíró (INK) színét, végül a képernyőgrafika (SCREEN) háttér (PAPER) és kiíró (INK) színét. Mindvégig csak 0-7-ig a megfelelő számértékekkel kell megadni a szükséges paramétert.

Print
Hard-copy-t hajt végre minden nyomtatóra, amely ismeri a COPY utasítást.

Keep
A szolgáltatás lehetővé teszi, hogy az aktuális képet SCREEN$ állományként közvetlenül kazettára mentsük. Kimentés előtt a program bekéri a file nevét.
Megjegyzés: Visszatöltéskor nem kell a VU-3D-nek a memóriában lennie. Nem ekvivalens a főmenüből meghívható "Save a data file" - opcióval, az csak egy adathalmazt ment ki kód formájában, ez pedig egy SCREEN$ típusú állományt.

Quit
Segítségével visszaléphetünk a DISPLAY menübe.
A főmenüből még lehetőség van a Change Colours opciót is meghívni, ebben az esetben a PICTURE menü Colour parancsának megfelelően kell eljárni.





A RAMBO második - 1770 byte-os - részének nyomtatott listája a következő:

```
0>

REM Protected by SPEEDLOCK
   0>BORDER 0: PAPER 0: INK 0: BRIGHT 1:
CLS: POKE 23624,0
   0>POKE (PEEK 23641+256*PEEK 23642),
PEEK 23649: POKE (PEEK 23641+256*PEEK
23642)+1,PEEK 23650
   0>POKE (PEEK 23633+256*PEEK 23634),
PEEK 23647: POKE (PEEK 23633+256*PEEK
23634)+1,PEEK 23648
   0>POKE 23662,PEEK 23618: POKE 23663,
PEEK 23619: POKE 23664, PEEK 23621
```

A sorszámozás 'különös', ettől eltekintve azonban mindjárt az a kérdés, hogy mitől indul el a következő részek betöltése. A gépi kód feltehetően további programsorokban, és a változóterületen található, de hogyan kerül rá a vezérlés? A listában szereplő rendszerváltozók átállítása ilyesmire nem alkalmas. Lehetséges tehát, hogy itt sem 'valódiak' a számértékek. Az előző részben közölt rövid rutin segítségével hamar kiderül, hogy a listában megjelölt sorban az ERR SP rendszerváltozó által megcímzett hiba-stack felső címe kerül át a változóterület elejére, így bármilyen programhiba - a normális programvégrehajtás befejezését is beleértve - az itt elhelyezett kódot indítja. (Megjegyezzük, hogy a változóterületen található a jól ismert 'Attempting to crack SPEEDLOCK can damage your sanity!' üzenet, azaz a SPEEDLOCK feltörésének kísérlete agykárosodást okozhat. Említsük meg a programozókat is, megérdemlik: DJL Software & D.Aubrey Jones.) Példánkban a VARS rendszerváltozó a #6226-os címet tartalmazza, itt kell tehát kezdenünk a visszafordítást. A kód eléggé meghökkentő: látszólag teljesen értelmetlen utasításokat is tartalmaz:

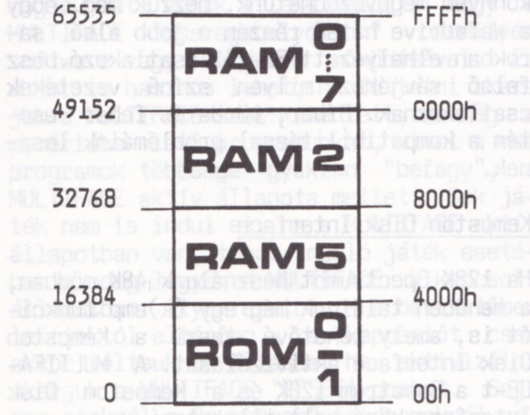
```
#6226   #6D          LD    L,L
#6227   #45          LD    B,L
#6228   #40          LD    B,B
#6229   #79          LD    A,C
#622A   #92          SUB   D
#622B   #ED#57       LD    A,I
#622D   #DD#62       LD    XH,D
#622F   #64          LD    H,H
#6230   #53          LD    D,E
#6231   #E0          RET   PO
#6232   #15          DEC   D
#6233   #15          DEC   D
#6234   #52          LD    D,D
#6235   #D9          EXX
#6236   #DD#68       LD    XL,B
#6238   #AA          XOR   D
#6239   #ED#62       SBC   HL,HL
#623B   #AB          XOR   E
#623C   #F3          DI
#623D   #6D          LD    L,L
#623E   #77          LD    (HL),A
#623F   #77          LD    (HL),A
#6240   #FD#84       ADD   A,YH
#6242   #FD#AD       XOR   YL
#6244   #39          ADD   HL,SP
#6245   #FD#62       LD    YH,D
#6247   #DD#54       LD    D,XH
#6249   #FD#AC       XOR   YH
#624B   #77          LD    (HL),A
#624C   #68          LD    L,B
#624D   #FD#69       LD    YL,C
#624F   #62          LD    H,D
#6250   #FD#63       LD    YH,E
#6252   #D9          EXX
#6253   #42          LD    B,D
#6254   #55          LD    D,L
#6255   #47          LD    B,A
#6256   #CB#1C       RR    H
#6258   #4F          LD    C,A
#6259   #46          LD    B,(HL)
#625A   #46          LD    B,(HL)
#625B   #CB#1D       RR    L
#625D   #59          LD    E,C
#625E   #D9          EXX
#625F   #A9          XOR   C
#6260   #59          LD    E,C
#6261   #4F          LD    C,A
#6262   #55          LD    D,L
#6263   #A8          XOR   B
#6264   #42          LD    B,D
#6265   #49          LD    C,C
#6266   #47          LD    B,A
#6267   #A7          AND   A
#6268   #57          LD    D,A
#6269   #41          LD    B,C
#626A   #4C          LD    C,H
#626B   #4C          LD    C,H
#626C   #59          LD    E,C
#626D   #DD #54      LD    D,XH
#626F   #FD #4D      LD    C,YL
#6271   #ED #4F      LD    R,A
#6273   #DD #7D      LD    A,XL
#6275   #FD #44      LD    B,YH
#6277   #68          LD    L,B
#6278   #59          LD    E,C
#6279   #6F          LD    L,A
#627A   #7F          LD    A,A
#627B   #6D          LD    L,L
#627C   #4A          LD    C,D
#627D   #40          LD    B,B
#627E   #ED #B0      LDIR
#6280   #FD #26 #FC  LD    YH, #FC
#6283   #48          LD    C,B
#6284   #4F          LD    C,A
#6285   #EE #05      XOR   05
#6287   #3B          DEC   SP
#6288   #3B          DEC   SP
#6289   #43          LD    B,E
#628A   #4D          LD    C,L
#628B   #6D          LD    L,L
#628C   #ED #52      SBC   HL,DE
#628E   #FD #2E #B7  LD    YL, #B7
#6291   #55          LD    D,L
#6292   #ED #5E      IM    2
#6294   #D9          EXX
#6295   #52          LD    D,D
```

```
#6296   #4F         LD    C,A
#6297   #5D         LD    E,L
#6298   #FD#E3      EX    (SP),IY
#629A   #41         LD    B,C
#629B   #4E         LD    C,(HL)
#629C   #44         LD    B,H
#629D   #ED#6A      ADC   HL,HL
#629F   #FD#54      LD    D,YH
#62A1   #FD#5D      LD    E,YL
#62A3   #EB         EX    DE,HL
#62A4   #42         LD    B,D
#62A5   #7E         LD    A,HL
#62A6   #55         LD    D,L
#62A7   #50         LD    D,B
#62A8   #ED#52      SBC   HL,DE
#62AA   #4A         LD    C,D
#62AB   #7B         LD    A,E
#62AC   #59         LD    E,C
#62AD   #4F         LD    C,A
#62AE   #55         LD    D,L
#62AF   #52         LD    D,D
#62B0   #53         LD    D,E
#62B1   #5F         LD    E,A
#62B2   #7E         LD    A,(HL)
#62B3   #44         LD    B,H
#62B4   #ED#57      LD    A,I
#62B6   #4D         LD    C,L
#62B7   #E1         POP   HL
#62B8   #52         LD    D,D
#62B9   #7C         LD    A,H
#62BA   #E8         RET   PE
#62BB   #E5         PUSH  HL
#62BC   #EB         EX    DE,HL
#62BD   #E4#08#30   CALL  PO,#3008
#62C0   #ED#5F      LD    A,R
#62C2   #AE         XOR   (HL)
#62C3   #77         LD    (HL),A
#62C4   #ED#A0      LDI
#62C6   #E0         RET   PO
#62C7   #3B         DEC   SP
#62C8   #3B         DEC   SP
#62C9   #E8         RET   PE
```

Ugyanennek a memóriaterületnek a karakteres kijelzése során értelmes szövegrészeket is találunk. A haszontalannak tűnő utasítások közé azonban ravaszul becsúszott egy két olyan is, amely nagyon lényeges a program működőképessége szempontjából. Az igazat megvallva minden olyan kísérletünk, amely a forgalomban lévő monitorprogramok szokásos szolgáltatásainak (töréspontok, lépésenkénti végrehajtás) segítségével próbálta nyomon követni a működést, kudarcot vallott. Néhány 'fogását' a kódnak megemlítenénk: Itt is megjelennek a nem standard Z-80 utasítások; letiltja a megszakításokat, majd 2-es interrupt módra vált; áthelyezi a stack-et; elrontja az IY és a HL' regiszterpárokat. Mindemellett (példánkban a # 627E címen) egy olyan LDIR utasítás található, amelyet nem előz meg a felhasznált BC, DE és HL regiszterpárok abszolút címmel történő feltöltése, ezeknek értéke a megelőző, jó néhány utasítás folyamán alakul ki. Példánkban HL= #63B8, DE= #63E0 és BC= #9B63 (hogy ez hogyan derül ki, arról majd később). Azaz az utasítás 'végigsöpri' maga felett a teljes memóriát, ez eleve kudarcra ítél minden itt elhelyezkedő segédprogramot. Néhány további - többé-kevésbé érthetetlen - utasítás után érünk el a SPEEDLOCK talán legrafináltabb részéhez (#62C0-#62C9), egy meglehetősen szokatlanul szervezett ciklushoz (véleményünk szerint ez okozza a feltörésnél a legnagyobb nehézséget).

Az itt szereplő regiszterek tartalma természetesen most sem olvasható ki közvetlenül a kódból. A HL által megcímzett byte-ot XOR-olja az R regiszterrel, visszateszi, majd azt áthelyezi a DE-ben lévő címre. HL és DE eggyel nő, és elérünk a RET PO utasításhoz. Ha BC már 1, akkor itt megtörténik a visszatérés, különben SP kettővel csökken, és a RET PE utasítást hajtja végre. Példánkban a ciklusba történő belépéskor HL=#5EFD, DE=#FCB7, BC=#0329, továbbá SP az #FCB7, SP-2 a #62C0 címre mutat. Tehát mindaddig a fenti ciklusmagot hajtja végre, amíg a #0329=809 XOR-olt byte-ot át nem helyezte az #FCB7 címre, majd ezen a címen folytatódik a végrehajtás.
A fenti címre kerülő kód tényleges visszaállításához szükségünk van az R regiszter értékére a ciklus elején. Ez a regiszter a Z-80 egy speciális regisztere: értéke minden utasításlehívási ciklusban eggyel nő modulo 128, azaz a legnagyobb helyiértékű bit nem változik, szabadon programozható. A program sok-sok utasítással a ciklus előtt (a #6271-es címen) állítja be értékét, a ciklusig végrehajtott utasítások már automatikusan változtatják. Példánkban a ciklusba történő belépéskor R=#CD, tehát az első byte (az#5EFD címen lévő) ezzel XOR-olva kerül az # FCB7 címre. Mivel maga a ciklusmag végrehajtása is változtatja az R regiszter értékét, a következő byte már egy más értékkel XOR-olódik. Ehhez elég tudni, hogy eközben R értéke tízzel nő, tehát a következő byte-ot a #CD+#0A=#D7 értékkel fogja XOR-olni. (Noha itt nem játszik szerepet, mégegyszer hangsúlyozzuk, hogy az összeadást modulo 128 kell végeznünk.)
Egyenként persze elég fáradságos volna 809 byte-ot állandóan változó számokkal XOR-olni. A következő részben arról szeretnénk írni, hogyan lehet a műveletet felgyorsítani, miként deríthetők ki a fent említett regiszterértékek, és mit tartalmaz a további kód.

A memória



A Z-80 mikroprocesszor egyidőben 65536 cím megkülönböztetését tudja elvégezni (0000-FFFFh). A hagyományos 48K Spectrum esetében az alsó 16K a ROM, míg a felette elhelyezkedő 48K memória a RAM.
A 128K Spectrum esetében a memóriatérkép hasonló, annyi különbséggel, hogy un. árnyék memóriaterületeket is találunk, amelyek megfelelő rutinok alkalmazásával lapozhatók. A 128K Spectrumban 2 ROM lap és összesen 10 RAM lap található, s minden egyes lap mérete 16K. Ebből adódóan öszszesen 131072 RAM és 32768 ROM byte-unk áll rendelkezésre, azaz összmemóriánk 163840 byte (160K). Mindettől függetlenül egyidőben akkor is csak 16K ROM és 48K RAM áll rendelkezésünkre.
A RAM 2 és a RAM 5 szelet mindig állandó, az itt található területre nem tudunk felüllapozni (4000h-C000h).
A RAM szempontjából érdekes terület a felső 16K (C000h-FFFFh), melyen belül újabb két típust különíthetünk el.
A RAM 4-7 a RAM 5-höz hasonlóan olyan memóriakörbe tartozik, amely részt vesz a video-kép kezelésében is, ezért kritikus idővel futó rutinok és ciklusok (pl. zene vagy töltő/mentő rutinok) itt nem futtathatók. Erre kiválóan alkalmas a RAM 0-3 terület, ahol ilyen gondba nem ütközünk.

Az egyes szeletek kijelölése a 32765 (7FFDh) I/O porton keresztül címezve oldható meg, a jobb oldali adatbitek jelölése a következő:

D0-D2 : RAM kiválasztás
D3 : képernyő kiválasztás
D4 : ROM kiválasztás
D5 : 48K üzemmód kapcsoló

A D2-D0 3 bites adat (0-7) választja ki a megfelelő RAM szeletet (C000h-FFFFh). BASIC programok írásakor a RAM 0 hagyományos tároló, míg a RAM 7-ben tárolódnak el a megfelelő változó pufferek. A D3 a képernyő-kapcsoló, 0 esetén a RAM 5 elejére, míg 1 esetén a RAM 7 elejére áll a képernyő-memória ill. rendszerváltozók. Ez utóbbi esetben a teljes 48K szabadon felhasználható (4000h-FFFFh), természetesen csak gépi kódból folyamatos adatok ill. rutinok tárolására.
A D4 a ROM kapcsoló. 0 esetén ROM 0, a szerkesztő ROM, míg 1 esetén ROM 1, a BASIC ROM lesz kiválasztva.
A D5 aktív állapota a rendszert a 48K üzemmódba irányítja át, megjegyeznénk, hogy innen már csak a RESET segítségével léphetünk vissza ismét a 128K módba.

A'Spectrum' utasítás

Ha 128K módban dolgozunk, és kiadjuk:

SPECTRUM (ENTER),

automatikusan áttérünk a 48K rendszerhez. 48K rendszerben nincs lehetőségünk 128K extra utasítások kihasználására, csak a 3 csatornás hang kezelését tudjuk megoldani, gépi kódú oldalról.
Ezért tapasztalhatjuk azt, hogy több 48K Spectrum játék (pl. THE TUBE, ELEVATOR ACTION, stb.) a 128K gépen 48K módban is 3 szólamú zenét produkál. A gyári kazettákon egyébként az ilyen játékokat már 48K/128K jelzéssel szokták ellátni.

Kick Boxing

Csak akkor működik ha a program egy fejléces CODE-ból áll (plusz a többi rész amit külön tölt be). A sérthetetlenséghez töltsük be a LOADER-t, majd RESET-eljük a gépet, és írjuk be a következő BASIC programot:

```
10 CLEAR 24999: LOAD"" CODE: POKE 55748,0: POKE 56741,0:
   RANDOMIZE USR 55561
```

Futtassuk a programot (RUN+ENTER) és indítsuk el a magnót.

A MULTIFACE Interface kompatibilitása

Beta-Disk Interface

A MULTIFACE kompatibilis a Beta 4.0 vagy magasabb verziókkal. A MULTIFACE-t a Spectrum és a Beta Interface közé kell elhelyezni. Ezen túlmenően szükséges egy kis változtatás a MULTIFACE belsejében. Ezt célszerű szakemberrel elvégeztetnünk, de ha él bennünk a barkács-szellem, és vállaljuk a kockázatot, hogy biztosan nem okozunk kárt készülékünkben akkor próbálkozhatunk az átalakítással. Ha a műanyag doboz hátlapját eltávolítottuk, találunk egy kis vezeték ívet, közvetlenül a joystick csatlakozó aljzat mellett. Ezt a vezetéket vágjuk át. Igaz, hogy ezután a joystick interface működésképtelen lesz, de ez a kis műtét szükséges, mert egyébként a Beta-Interface működésében állnak elő zavarok. Természetesen a legcélszerűbb megoldás egy kapcsolót beépítenünk az átvágás helyére, így a kapcsoló ON állásában működik a joystick Interface, OFF állásban pedig a Beta Interface.

Microdrive

A MULTIFACE-t az Interface I mögé kell illesztenünk. A MULTIFACE kompatibilis az Interface I. összes verziójával.

OPUS Discovery

A MULTIFACE-t a Spectrum és az Opus Discovery közé kell beiktatnunk. Figyelem! Az Opus Discovery oldalán továbbmenő 'élcsatlakozó' a MULTIFACE-szel párhuzamosan nem használható.

Magnetofon

A MULTIFACE bármikor képes minden megkötöttség nélkül programot kimenteni kazettás magnetofonra. A kimentés a hagyományos vonalon történik. A kazettáról betöltött program bármikor átmenthető egyéb háttértároló eszközre (pl. microdrivera).

Wafadrive

A MULTIFACE a Spectrum és a Wafadrive közé, vagy a Wafadrive mögé is csatlakoztatható. A Wafadrive a MULTIFACE által automatikusan kiválasztott periféria, ezért vezérlése is automatikusan történik. Némelyik Wafadrive modell nem teljesen kompatibilis a MULTIFACE-szel, erről könnyen meggyőződhetünk, nézzük meg, hogy a Wafadrive hátsó részén a jobb alsó sarokban elhelyezett RS-232 csatlakozó busz felső sávjához milyen színű vezetékek csatlakoznak. Bíbor, vörös és fehér esetén a kompatibilitással problémáink lesznek.

Kempston Disk Interface

Ha 128K Spectrumot használunk 48K módban, a menüben találunk még egy (k)emp funkciót is, amely lehetővé teszi a Kempston Disk Interface aktivizálását. A MULTIFACE-t a Spectrum 128K és a Kempston Disk Interface közé kell illeszteni.

Javaslatok a MULTIFACE felhasználásához

A MULTIFACE varázsgombját bármikor megnyomhatjuk, függetlenül attól, hogy a program milyen típusa található a memóriában, vagy attól, hogy begépelt programról, vagy betöltött programról van-e szó. A MULTIFACE a memóriatartalmat tömörített RAM formában menti ki. A kimentés mindig 4 elkülönített részben valósul meg:

- betöltő BASIC; - képernyő;
- főkód; - műveleti kód

Minden program esetén rendelkezésünkre áll egy teljes törlési lehetőség is. Hasznos, hogy a BREAK funkciót is aktivizálhatjuk bármelyik periféria működése alatt. Ha hiba fordul elő, visszatérünk a menühöz. Sajnos különböző rendszerek felhasználása különböző hibacsapdákat eredményezhet.
A Beta Disk Interface + Drive, valamint a Wafadrive kijelzi azoknak a file-oknak a nevét, amelyeket már kimentettünk. Mindkét rendszer megengedi, hogy felülírjuk az előzőleg kimentett file-t, vagy a file nevét átnevezzük.
Az Opus Discovery automatikusan felülírja az azonos néven előzőleg kimentett file-t.
A három rendszer közül egyik sem kezdi el a kimentést, ha a kimentés kezdeti pillanatában kevesebb hely áll rendelkezésre a háttértárolón, mint amekkora a program tömörített formájának a hossza.
A microdrive hibaüzenetek pl. 'Microdrive not present', 'File already exists' az

ismert I/O hibaüzeneteknek megfelelőek.
A MULTIFACE oldalán elhelyezett kapcsoló a MULTIFACE-t "láthatatlanná" teszi. Ez megszünteti annak a lehetőségét, hogy a MULTIFACE "összegabalyodjon" bármilyen software vagy hardware működésével. Ez akkor is hasznos lehet, ha futtatni akarjuk a MULTIFACE-szel kimentett, de még a memóriában lévő programunkat, mivel a mai programok többsége gyakran "befagy", a MULTIFACE aktív állapota mellett. Sok játék nem is indul el, ha a MULTIFACE aktív állapotban van, tehát induló játék esetében mindig legyen a MULTIFACE kikapcsolt állapotban. Ilyen esetben - a bevezetőben leírtaktól eltérően - a kapcsolót csak akkor váltsuk ON állásba, ha aktivizálni akarjuk a MULTIFACE 'RESET' gombot. Sajnos ezeknél a "gubancos" programoknál nagyon nagy az esélye annak, hogy (r)eturn, vagy (e)xit esetén "befagy" a rendszer. Ekkor nincs más hátra, újra be kell tölteni.

A MULTIFACE programozása

A MULTIFACE 8K szabadon felhasználható RAM-mal rendelkezik, amely átmeneti pufferként alkalmazható, ezen túl a rendszer vezérlését egy 8K ROM (EPROM) végzi el.
A plusz 8K RAM-ot felhasználhatjuk saját gépi kódú rutinok (esetleg monitor/disassembler programok) tárolására, vagy egyéb adatok megőrzésére is. BASIC program itt nem tárolható.
A RAM aktivizálása előtt a memóriát lapoznunk kell. A lapozás csak gépi kódban oldható meg, a BASIC IN és OUT nem működik.

Az itt ismertetett blokk-mozgató rutin példát mutat a memória lapozására:

```
 10 RESTORE
 20 GO TO 50
 30 LET H=INT (1/256)
 40 LET L=1-H * 256: RETURN
 50 INPUT "Start cím:";i:GO SUB 30
 60 LET SH=H: LET SL=L
 70 INPUT "Cél cím:";i:GO SUB 30
 80 LET CH=H: LET CL=L
 90 INPUT "Blokk hossz:";i:GO SUB 30
100 LET BH=H: LET BL=L
110 FOR i=23296 TO 23313
120 READ a: POKE i,a: NEXT i
130 RANDOMIZE USR 23296
140 DATA 243,219,159,33,SL,SH
150 DATA 17,DL,DH,1,BL,BH
160 DATA 237,176,219,31,251,201
```

Most nézzünk egy példát a konkrét adatmozgatásra. Rajzoljunk valamit tetszés szerint a képernyőre, és ezt mozgassuk át a MULTIFACE RAM területre.
Ha a képünk előállt, GO TO 1 (nem RUN!), ezt követően adjuk meg a bemenő paramétereknek a következőket: Start cím: 16384; Cél cím: 8192; Blokk hossz: 6912.
Töröljük a képernyőt és futtassuk újra a programot (ha most RUN-nal indítjuk, a törlést automatikusan elvégzi). Irjunk startcímnek 8192-t, célcímnek 16384-et, blokkhossznak 6912-t és a kép ismét megjelenik a képernyőn.

A plusz RAM természetesen ezen túlmenően számos lehetőséget biztosít nekünk. A legcélszerűbb, ha ide töltünk assemblert, monitort, vagy egyéb toolkit-et.
A NEW nem törli a 8K RAM területet, és a tartalma mindaddig megőrződik, amíg a MULTIFACE feszültség alatt van.

A 8K RAM tartalma közvetlenül nem menthető ki, azt először át kell mozgatnunk a Spectrum RAM szabad területére, s a kimentés onnan már elvégezhető.

A MULTIFACE operációs rendszere lehetővé teszi közvetlen ugrások végrehajtását is a varázsgomb segítségével. A varázsgomb leállítja bármelyik program futását, amelyik éppen a memóriában található. Alaphelyzetben ilyenkor a főmenü jelentkezik be a képernyő alsó sorában. Ez a manipuláció kb. 2500 byte-ot használ fel a 8K RAM területből. A gép a BASIC-hez való visszatéréshez a 11743-11746 címeket használja fel. Ehelyett természetesen saját visszatérő rutinokat is írhatunk, pl. visszatérés előtt még írja ki nekünk a regiszterek aktuális állapotát, a megszakítási státuszt, stb.
Az előbb említett direkt ugró rutin belépési címe a 8192/8193 címeken tárolódik. A 8195/8196/8197 címek speciális azonosító kódot tartalmaznak. Ez RUN esetén 82/85/78 (decimális). A 8194. cím tartalma jelen esetben lényegtelen.
Bármikor, ha megnyomjuk a varázsgombot, a direkt ugrás a megadott címre történik, és nem a MULTIFACE menü jelenik meg.
A MULTIFACE menübe úgy térhetünk vissza, hogy a varázsgombot és a BREAK billentyűt egyidőben nyomjuk meg.
Normál módban a MULTIFACE használja a 8210-10753 címeket, mint puffert, kimentés közben pedig a 8210-12722 terület is foglalt. Ilyen esetben az éppen ott lévő adataink felülíródnak.

Egyéb disk rendszerek felhasználása

A MULTIFACE teljesen kompatibilis a Discovery és a Beta valamint 128K Spectrum esetén a Kempston Disk illesztőkkel, csak a menüben kell kijelölnünk a megfelelő opciót.

Egyéb disk rendszerek is felhasználhatók, de ez esetben előbb ki kell menteni a programunkat szalagra, majd el kell végezni a módosítást az új disk rendszerrel való felhasználáshoz.
Ehhez szükséges két segédprogram, amelyek a mentő/töltő műveleteket végzik el a disk-re.
KEMPSTON rendszer esetén, ha a (k)emp nem jelenik meg a menüben, előbb mentsük ki a programot kazettára, áramtalanítsuk a gépet, csatoljuk szét a MULTIFACE rendszert, illesszük a gépre a Kempston Interface-t, kössük rá a disk-et, majd futtassuk a következő programot:

```
10 CLEAR VAL "24751"
20 LOAD"" CODE: PRINT# 4:
   SAVE "3" CODE VAL "24752",x
30 LOAD"" CODE: PRINT# 4:
   SAVE "2" CODE VAL "16384",y
40 LOAD"" CODE VAL "16464":
   PRINT# 4: SAVE "1" CODE
   VAL "16464",VAL "1906"
```

Az x változó értéke a 20. sorban a főkód hossza, maximum 40783 byte. Az y változó értéke a 30. sorban a képernyő hossza, maximum 6912 byte. Mindkét blokk tömörített és fejléc nélküli. A 3. blokk mindig 1906 byte hosszú. BASIC program esetén az eljárás ismert.

Más rendszerek esetén a PRINT# 4 utasítások helyett a megfelelő csatorna adatot kell beállítani.

A Disk-ről való visszatöltéshez is szükséges egy BASIC program. Kempston Interface-hez írjuk be és futtassuk a következőt:

```
10 CLEAR VAL "24751": PRINT# 4:
   LOAD"3" CODE
20 PRINT# 4: LOAD"2" CODE
30 RANDOMIZE USR VAL "24792":
   PRINT# 4: LOAD"1" CODE
40 RANDOMIZE USR VAL "20795"
```

Természetesen egyéb disk rendszereknél a PRINT utasítás mellett a csatornaszám módosítható.

Equinox

POKE: Meg kell találnunk a PETE feliratú vörös négyzetet, fel kell vennünk, majd vissza kell vinnünk a START-helyre, itt fel a bal felső sarokba és egyszerre megnyomni a FEL,R,N,C billentyűket.

I Ball

File-térképe: BASIC/6916/20000/20536/1704
Az örökélethez, és végtelen időhöz töltsük be a LOADER-t: MERGE"", majd írjuk be a következő BASIC programot:

```
30 FOR i=23841 TO 23853: READ a: POKE i,a: NEXT i
40 DATA 175,32,75,193,32,16,192,61,55,229,195,86,5
```

Futtassuk a programot GO TO 30 -cal majd RUN+ENTER és indítsuk el a magnetofont.

Shockway-Raider

A program file-térképe a következő: 170/34724/6052/1968
Az örökélethez töltsük be a LOADER-t, majd RESET-eljük a gépet, és írjuk be a következő BASIC programot:

```
10 CLEAR 24791: LOAD"" CODE: LOAD"" CODE: RANDOMIZE USR 24830:
   LOAD"" CODE 16464
20 POKE 46119,0: RANDOMIZE USR 24833
```

Futtassuk a programot (RUN+ENTER) és indítsuk el a magnót.

III. ÖSSZEADÓ UTASÍTÁSOK

Az összeadás fogalmát sokféleképpen értelmezhetjük. A Z-80 utasításkészletében elhelyezett összeadó utasítások csak az abszolút bináris aritmetika szabályai szerint képesek műveleteket végrehajtani, ezt pedig úgy végzik el, hogy egy megadott számhoz egy egy másik számot, ill. egy regiszter, egy regiszterpár vagy egy megcímzett memóriarekesz tartalmát adjuk hozzá. Ebből is kiderül, hogy a kezelhető adatok nagysága kötött, 1 ill. 2 byte-os számok kezelésére nyílik lehetőség, azaz max. hFFFF (65535) nagyságú érték kezelhető. Ez a tény azt világítja meg a számunkra, hogy ezek az összeadási műveletek elsődlegesen gépi kódú programjaink segédelemei lehetnek, matematikai jellegű számításokhoz, - ahol feltételezhető a hFFFF (65535) értéknél nagyobb egész illetve a valós számok felhasználása - célszerű külön lebegőpontos kalkulátort felhasználni. Ilyen lebegőpontos kalkulátort elhelyeztek a gép ROM-jában is, erre majd a későbbiekben visszatérünk.
A Z-80 összeadó utasításait három külön csoportra bonthatjuk:

1. ADD utasítások

Ezek az utasítások regiszterek, vagy regiszterpárok tartalmát összegzik, túlcsordulás esetén a végeredmény zérusba fordul át, és az átviteli jelzőbit felveszi az 1 értéket.

```
      198,n        C6  N        ADD  A,n      ;regiszterek összegzése
        135            87        ADD  A,A      ;konstanssal, vagy egy
        132            84        ADD  A,H      ;másik regiszter
        133            85        ADD  A,L      ;tartalmával.
        128            80        ADD  A,B
        129            81        ADD  A,C
        130            82        ADD  A,D
        131            83        ADD  A,E
        134            86        ADD  A,(HL)
  221,134,x        DD 86 X      ADD  A,(IX+x)
```

ahol n (hN) a konstans adatbyte, ill. x (hX) az eltolási kód értéke.

Az utasítások mechanizmusát BASIC-ben mutatjuk be:

```
        128            80        ADD  A,B
```

működése:

```
10 LET A=A+B
20 IF A>255 THEN LET ÁTV=1: LET A=A-256
```

ahol A és B jelzi a megfelelő regisztereket. Ha az összeadás eredménye meghaladja a 255-öt, az átviteli jelzőbit 1-re vált, és a végeredményt töröljük.

```
         41            29        ADD HL,HL     ;regiszterpárok
          9            09        ADD HL,BC     ;tartalmának
         25            19        ADD HL,DE     ;összegzése.
         57            39        ADD HL,SP
     221,41         DD 29        ADD IX,IX
      221,9         DD 09        ADD IX,BC
     221,25         DD 19        ADD IX,DE
     221,57         DD 39        ADD IX,SP
```

Az IY regiszterpárt tartalmazó utasítások esetén az IX helyére IY-t, a 221 helyére 253-at, DD helyett pedig FD-t kell írnunk.

A mechanizmus BASIC szemléltetése:

```
  25          19            ADD HL,DE
```

működése esetén

```
10 LET HL=HL+DE
20 IF A>65535 THEN LET ÁTV=1: LET HL=HL-65535
```

ahol HL és DE jelzi a megfelelő regisztereket. Ha az összeadás eredménye meghaladja a 65535-öt, az átviteli jelzőbit 1-re vált, és a végeredményt töröljük.

2. INC utasítások

Ezek az utasítások azt teszik lehetővé, hogy 1-et adjunk hozzá egy 8 bites regiszter, egy megcímzett memóriarekesz, vagy egy 16 bites regiszterpár tartalmához.

```
      60        3C         INC A        ;regiszter, vagy egy
      36        24         INC H        ;memóriarekesz
      44        2C         INC L        ;tartalmának
       4        04         INC B        ;növelése 1-gyel
      12        0C         INC C
      20        14         INC D
      28        1C         INC E
      52        34         INC (HL)
221,52,x     DD 34 X       INC (IX+x)
253,52,x     FD 34 X       INC (IY+x)
```

ahol x (hX) az eltolási kód értéke.

A működés BASIC-ben szemléltetve a következő:

```
      60        3C         INC A
```

esetén

```
10 LET A=A+1
20 IF A>255 THEN LET A=0
```

ahol 'A' az A regisztert jelzi.

```
      35        23         INC HL       ;regiszterpár tartalmának
       3        03         INC BC       ;növelése 1-gyel
      19        13         INC DE
      51        33         INC SP
  221,35     DD 23         INC IX
  253,35     FD 23         INC IY
```

A működés menete BASIC-ben a következő:

```
      35        23         INC HL
```

esetén

```
10 LET HL=HL+1
20 IF HL>65535 THEN LET HL=0
```

Megjegyzés: Az INC utasítások még túlcsordulás esetén sem befolyásolják az átviteli jelzőbit állapotát !

3. ADC utasítások

Ezek az utasítások hasonló mechanizmus szerint működnek, mint az ADD utasítások, annyi különbséggel, hogy összegzéskor a végeredményhez még hozzáadjuk az átviteli jelzőbit tartalmát is.

```
    206,n     CE N         ADC A,n       ;regiszter összegzése
    143       8F           ADC A,A       ;konstanssal, vagy egy
    140       8C           ADC A,H       ;másik regiszter
    141       8D           ADC A,L       ;tartalmával, ill.
    136       88           ADC A,B       ;az átviteli jelzőbit
    137       89           ADC A,C       ;tartalmával.
    138       8A           ADC A,D
    139       8B           ADC A,E
    142       8E           ADC A,(HL)
221,142,x     DD 8E X      ADC A,(IX+x)
253,142,x     FD 8E X      ADC A,(IY+x)
```

ahol n (hN) a konstans adatbyte, ill. x (hX) az eltolási kód értéke.

A mechanizmus működése BASIC-ben:

```
    136       88           ADC A,B
```

esetén

```
10 LET A=A+B+ÁTV
20 IF A<256 THEN LET ÁTV=0
30 IF A>=256 THEN LET A=A-256: LETÁTV=1
```

azaz A és B jelzi az aktuális regisztereket, ÁTV pedig az átviteli jelzőbitet. Ha az A regiszterben tárolt végeredmény értéke 256-nál kevesebb, úgy az átviteli jelzőbitet töröljük, egyébként 1-be állítjuk és az eredményt módosítjuk.

```
    237,106   ED 6A        ADC HL,HL     ;regiszterpárok tartalmának
    237,74    ED 4A        ADC HL,BC     ;és az átviteli jelzőbitnek
    237,90    ED 5A        ADC HL,DE     ;az összegzése
    237,122   ED 7A        ADC HL,SP
```

A mechanizmus BASIC-ben:

```
    237,74    ED 4A        ADC HL,BC
```

esetén

```
10 LET HL=HL+BC+ÁTV
20 IF HL<65535 THEN LET ÁTV=0
30 IF HL>=65535 THEN LET ÁTV=1: LET HL=HL-65535
```

Láthatjuk, hogy a mechanizmus működése megegyezik, csak az intervallum nagyobb mértékű.

Az összeadó utasításokkal kapcsolatban végezzünk el néhány mintapéldát az eddig megismert utasítások felhasználásával.
Az előző részben 'bonyolult' gépi kódú programunk eredményeképpen már meg tudtunk jelentetni egy kis vonalszakaszt a képernyő bal felső sarkában. Most próbáljuk meg ezt a feladatot tovább 'bonyolítani'.
Az egymás melletti 4 db. karakter felső byte-ját töltsük fel 1-es bitekkel, azaz egy 4 karakteres hosszú vonalat húzunk. A rutint tetszés szerint helyezzük el a memóriában, most mintapéldánkban az 50000. decimális címtől tettük meg ezt.

8. GÉPI KÓD TANFOLYAM

```
50000  33,0,64    21 00 40    LD   HL,16384    ;HL-ben a képernyő bal
                                               ;felső karakterének a
                                               ;címe.
50003  48,255     36 FF       LD   (HL),255    ;A bal felső sarokban
                                               ;megjelenik a vonal.
50005  35         23          INC  HL          ;Egymás után 4 lépésben
50006  48,255     36 FF       LD   (HL),255    ;léptetjük a képernyő-
50008  35         23          INC  HL          ;cím értékét és megje-
50009  48,255     36 FF       LD   (HL),255    ;lenítjük a vonalat.
50011  35         23          INC  HL
50012  48,255     36 FF       LD   (HL),255
50014  201        C9          RET
```

Az INC HL utasítás helyett alkalmazhattuk volna az ADD HL,BC utasítást is, de ez esetben a rutin elején meg kellett volna adni az LD BC,1 utasítást. Az ADD utasítás nagyobb léptékű emelkedést tehet lehetővé, a most említett szerkezeti változtatás mellett töltsük fel a képernyő bal-felső karakter-pozíciójának 4 egymás alatti byte-ját 255-tel, vagyis egy féltömör grafikus karaktert kapunk eredményül. Tudjuk, hogy az egymás alatti byte-ok címkülönbsége 256 byte, vagyis BC-ben ezt kell megadnunk.

```
50000  33,0,64    21 00 40    LD   HL,16384    ;a bal felső (0,0)
50003  1,0,1      01 00 01    LD   BC,256      ;karakter felső 4
50006  48,255     36 FF       LD   (HL),255    ;byte-jának bitjeit
50008  9          09          ADD  HL,BC       ;1-gyel töltjük fel.
50009  48,255     36 FF       LD   (HL),255
50011  9          09          ADD  HL,BC
50012  48,255     36 FF       LD   (HL),255
50014  9          09          ADD  HL,BC
50015  48,255     36 FF       LD   (HL),255
50017  201        C9          RET
```

Az itt ismertetett két mintapélda alapján is jól látható, hogy minél több byte-ot kell megcímeznünk, annál hosszabb a program, vagyis ez a módszer nem a legszerencsésebb. Ehelyett célszerű a ciklusutasítást (DJNZ) alkalmazni, amellyel csak a későbbiekben fogunk megismerkedni.

Tartalomjegyzék:

CÍMÜNK: SPECTRUM VILÁG

BUDAPEST - 3

Postán maradó

1300

Felelős kiadó: Rucz Lajos, Székely László· K. F. eng.-sz.: 53268
10 000 — 807417 Pátria Nyomda
Felelős vezető: Vass Sándor vezérigazgató

ENTERPRISE

A Sinclair ZX Spectrum rendkívül sikeres megjelenését követően a LOCUMALS - egy hong kong-i kereskedelmi vállalat - úgy döntött, eljött az ideje, hogy betörjenek a nagy haszonnal kecsegtető home computer piacra. 1982 szeptemberében a United Commercial Bank a LOCUMALS megbízásából felkereste az Intelligent Sofware Ltd.-t, és ajánlatot tett egy teljesen új számítógép kifejlesztésére. Az IS előtt a megrendelő neve titok maradt, ennek ellenére vállalta a megbízást, és a következő hónap elejére a DPC projekt elindult.

Amíg az IS az alapsoftware-t fejlesztette, Geoff Hollington és Nick Oakley a külső megjelenést formálták, Nick Troop és Dave Woodfield a gép lelkét alkotó 2 db. 72-lábú 'soft' ULA megtervezésével lett megbízva.

Ennek a merész fejlesztésnek az eredményeként látott napvilágot az **ENTERPRISE** számítógép.

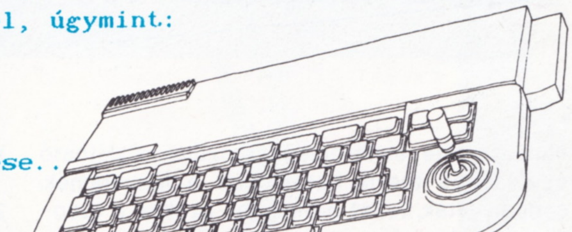
Bár hazánkba kissé megkésve érkezett, az érdeklődő felhasználók szívesen fogadták.

Hogy miért írjuk ezeket a sorokat annak több oka is van:

Először is az **ENTERPRISE** mikrogép lelkét szintén egy **Z80** processzor alkotja. Másodszor, az **ENTERPRISE** egy magyar találmány segítségével ZX Spectrummá válik, így a Spectrum software-ek 80 %-a futtatható rajta. Harmadszor, az **ENTERPRISE** körül óriási információs z'űr' kavarog.

Ezen próbálunk segíteni a jövőben rendszeres **ENTERPRISE**-ra vonatkozó információk közlésével, úgymint:

- Híd a Spectrum felé...
- Az ENTERPRISE 128
- Az ENTERPRISE képernyő-kezelése...
- A Hisoft DEVPAC kezelése
- stb.



SPECTRUMOSOK FIGYELEM!



Szolgáltatás jelleggel **program-kazetta küldő szolgálatot** szeretnénk indítani a közeljövőben.

- **önálló programok** megjelölésével
- **kollekciók** összeállításával

A szolgáltatás beindulásának időpontjáról, illetve a megjelölhető programokról és kollekciókról időben tájékoztatót fogunk közölni a SPECTRUM VILÁG-ban.
Addig is **várjuk** tervezett tevékenységünkről **véleményüket, javaslataikat**!

'SV'

VÉGRE!
EGY ÉRTELMES JÁTÉK,
amely igaz, hogy tanít, de a játszás örömével együtt.



A panelen 24-féle elektronikus készülék alakítható ki. Élvezetesebbé teszik használatát a fényelemek. Lehet csupán játék, lehet betekintés azokba a tudományokba, amelyek ismerete a közeljövőben meghatározza életünket. Tartalma:

Játékok – ugyanakkor valóságos készülékek.
- Rádió vevők;
- rádió adók;
- számítógép alapáramkörök;
- elektronikus cica, csipogó, morse

és mindezek nap- és fényenergiával működtetve vagy vezérelve.

Kapható:

2C Áruház, XIII. Balzac u. 35.
Ápisz, XI. Budafoki út 7.
Fókusz könyváruház, VII. Rákóczi út 14.
MIGÉRT bemutatóterem, VIII. Rákóczi út 57.
MOZAIK üzletház, V. Petőfi Sándor u. 10.
OTTHON áruház – műszaki osztály
TECHNIKA könyvesbolt, XI. Bartók B. út 15.
ÚTTÖRŐ áruház – műszaki osztály

valamint Tatabányán
CENTRUM áruház – műszaki osztály – AlfaDat GT

Megrendelhető illetve megtekinthető: LSI ATSz, 1033 Budapest III. Hévízi út 6/e.

Ára: 1994.– Ft